# INIȚIERE ÎN ACCESS



4 Colecția: INIȚIERE ÎN CALCULATOARE

EDITURA ARVES

*Redactor: Sorin Matei*

**ISBN 973 – 85773 – 1 – 4**

**Tiparul executat la S.C. LIBRA COM S.R.L. Craiova**
**Imprimeria Editurii MJM**
**Tel.: 051/419661, Fax: 051/419661; Mobil: 095512223**
**IMPRIMAT ÎN ROMÂNIA**

Cornel IGNAT Carmen PETRE

# INIȚIERE ÎN ACCESS

EDITURA ARVES

În „Colecția: Inițiere în calculatoare" **au apărut**::
Inițiere în utilizarea calculatoarelor
Inițiere în WORD
Inițiere în EXCEL
Inițiere în ACCESS
Inițiere în INTERNET, E-MAIL, CHAT

În „Colecția: INFORMATICA" **au apărut:**
Teorie şi aplicații în PASCAL – clasa a IX – a
Teorie şi aplicații în PASCAL – clasa a X – a
Metodica predării informaticii şi tehnologiei informației în liceu

**Au mai apărut:**
Consiliere şi orientare şcolară
Caietul dirigintelui

**În curs de apariție** în „Colecția: Inițiere în calculatoare":
Inițiere în PowerPoint
Inițiere în FOXPRO

**Pentru informații şi comenzi:**
Editura "Arves"
Tel./ Fax: 051-419661
Mobil Matei Sorin: 091229348
Adresa: Str. Mărăşeşti nr. 44, Craiova, cod. 1100, Dolj

# Cuprins

# Introducere

Prelucrarea informației necesită de cele mai multe ori un volum foarte mare de calcule şi uneori aplicarea unor formule inaccesibile programelor de calcul tabelar. Apar de asemenea probleme foarte greu de rezolvat atunci când se impune preluarea şi interpretarea datelor dintr-o grilă bi sau tridimensională. În aceste cazuri se impune de la sine folosirea bazelor de date.

Conceput pentru prelucrarea datelor dintr-o bază de date, Microsoft Access 97 aduce îmbunătățiri substanțiale versiunilor mai vechi de Access (1.x, 2.0, 7.0/95, sau 8.0/97) solicitând în acelaşi timp destul de puține resurse din parte maşinii pe care lucrează.

**Resursele solicitate** de Microsoft Access 97:

Procesor 486 sau mai mare.

Sistem de operare Windows 95 sau versiuni superioare.

Minim 8 Mb memorie RAM (funcționează şi cu 4Mb).

Monitor VGA – 256 culori (minim)

60-167 Mb liberi pe hard disk

În condițiile unei nevoi tot mai acute de programe, Access 97, prin facilitățile pe care le pune la dispoziția utilizatorului şi programatorului şi prin simplitatea înțelegerii, constituie mediul de programare ideal pentru crearea propriului software.

Cartea de față vă oferă soluții pentru înțelegerea realizării şi funcționării bazelor de date, ajutându-vă să construiți propria aplicație şi propria bază de date. Exemplul folosit în această carte permite realizarea unei aplicații pentru centralizarea situației şcolare a elevilor din rețeaua Ministerului Educației Naționale.

Chiar dacă nu s-a insistat asupra elementelor de programare în cod VBA, asistenții Wizard încorporați oferă suficiente posibilități pentru crearea unei aplicații şi a unei baze de date puternice şi performante.

# Microsoft Access 97

Access 97 ca parte integrantă a pachetului Microsoft Office 97 Professional Edition vine în completarea facilităților oferite de lucrul cu foile de calcul tabelar, asigurând stocarea şi prelucrarea unui volum foarte mare de date şi în acelaşi timp conlucrarea cu celelalte programe incluse în acest pachet, dar şi colaborarea cu alte tipuri de baze de date folosite în prezent (Fox Pro, Dbase, Paradox, etc.).

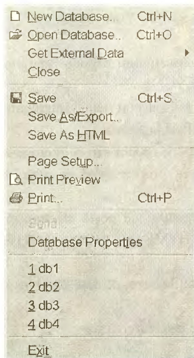
Avantajele folosirii bazelor de date Access 97 , prin comparație cu foile de calcul tabelar constau în:

1. Implementarea sistemului de relații.
2. Organizarea datelor .
3. Restructurarea datelor în tabele relaționate.
4. Folosirea eficientă a memoriei.
5. Asigurarea consecvenței în folosirea denumirilor.
6. Posibilitatea memorării oricărui tip de informație.
7. Folosirea mai eficientă a memoriei.
8. Navigare în rețeaua Internet şi Intranet.
9. Colaborarea cu alte aplicații.
10. Folosirea elementelor de personalizare a aplicațiilor.
11. Folosirea filtrelor.
12. Programarea în Visual Basic.
13. Securitatea şi administrarea bazelor de date.

Acest program vă va permite, după ce veți ajunge să stăpâniți tehnicile de bază ale prelucrării datelor, să construiți şi să distribuiți propriile aplicații.

# PREZENTAREA MENIURILOR

## Meniul File (Fişiere)




**New Database...** – permite crearea unei noi baze de date.

**Open Database...** - permite deschiderea unei baze de date.

**Get External Data** cu articolele de meniu **Import** şi **Links** permite:

**Import:** Copierea datelor dintr-un fişier text sau dintr-un tabel al unei alte baze de date într-un tabel al bazei de date curente.

**Links:** Crează o legătură, în baza de date curentă, la un tabel al unei alte baze de date.

**Close** – Închide fereastra curentă. Dacă aţi efectuat modificări într-unul din obiectele bazei de date şi nu l-aţi salvat încă, Access 97, vă va propune efectuarea salvării acestuia.

**Save** – permite salvarea aspectului unui tabel, formular, raport sau a unei interogări, sau a aspectului sau conţinutului unui modul.

**Save As/Export**

- Creează (prin salvare) o copie cu nume diferit a obiectului selectat în baza de date curentă.

- Exportă rezultatul prelucrării obiectului selectat într-un fişier extern (în format .txt sau .rtf) sau într-o bază de date externă (Microsoft Access sau Paradox).

- Converteşte macro-ul selectat într-un nou modul Visual Basic.

**Save As HTML** – permite exportul obiectului selectat sau al conţinutului acestuia într-un format specific paginilor Web (.html, .idc/.htx şi .asp).

**Page Setup** – permite controlul caracteristicilor paginii (margini, dimensiuni, orientare) unui formular sau raport.

**Print Preview** - afişează obiectul activ sau selectat (tabel, interogare, formular, raport) în forma în care acesta va fi afişat la tipărire.

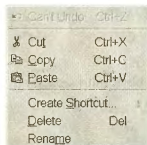
**Print** – Tipăreşte un tabel, formular, raport, modul sau o interogare. Afişează dialogul Print permiţând modificarea proprietăţilor şi setărilor imprimantei.

**Send** – Trimite prin intermediul programului de poştă electronică, rezultatele prelucrate într-unul din formatele: HTML, MS-DOS text, RTF, sau Excel.

**Database Properties** – Afişează proprietăţile bazei de date astfel încât să poată fi modificate de utilizator.

**Exit** – Închide aplicaţia Access şi baza de date curentă.

## Meniul Edit (Editare)




**Undo** - Creează o comandă cu efect contrar ultimei acţiuni a utlizatorului. Numele acestei comenzi depinde de numele acţiuni întreprinse de utilizator (Exemplu: Undo Cut, Undo Move, etc.). Dacă ultima acţiune întreprinsă de utlizator nu permite creearea unei acţiuni cu efect contrar, numele acestei comenzi apare în meniul edit sub denumirea Can't Undo.

**Cut** - Şterge obiectul selectat şi îl plasează în clipboard astfel încât să poată fi inserat în altă parte. Acţiunea inversă acestei comenzi este comanda Undo Cut.

**Copy** - Copiază obiectul selectat în clipboard (Exemplu: o înregistrare sau un control) astfel încât acesta să poată fi introdus în altă parte.Acţiunea inversă acestei comenzi este comanda Undo Copy.

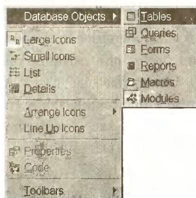
**Paste** - Introduce un obiect capturat în clipboard în baza de date curentă (O înregistrare sau un control). Acţiunea inversă acestei comenzi este Undo Paste.

**Create Shortcut** - Permite crearea unei scurtături către un obiect din baza de date.

**Delete** - Şterge definitiv obiectul selectat fără a-l plasa în clipboard. În fereastra "Relationship" (Relaţii), ştergerea unei linii de relaţionare echivalează cu ştergerea relaţiei dintre două tabele.

**Rename** - Redenumeşte obiectul selectat din baza de date. Acţiunea inversă acestei comenzi este Undo Rename.

## Meniul View (Vizualizare)




**Database Objects - Tables** Afişează în fereastra bazei de date lista tabelelor din baza de date curentă.

**Database Objects - Queries** Afişează lista interogărilor din baza de date curentă.

**Large Icons** Afişează obiectele bazei de date folosind iconiţe mari înaintea numelui acestora.

**Small Icons** Afişează obiectele bazei de date folosind iconiţe mici înaintea numelui acestora.

**Line Up Icons** Aranjează într-o rețea ortogonal obiectele din fereastra bazei de date.

**Properties** Afişează proprietățile obiectului selectat, iar în situația în care nici un obiect ne este selectat, proprietățile bazei de date .

**Code** Afişează codul sursă care lucrează în spatele formularului sau raportului selectat.

**Toolbars - Bare de instrumente -** Afişează sau ascunde bare de instrumente. Pentru a afişa o bară de instrumente, selectați caseta de selectare de lângă numele acesteia. Pentru a ascunde o bară de instrumente, deselectați caseta de selectare.

**Toolbars – Customize - Particularizare** Particularizează butoanele barei de instrumente, comenzile meniurilor şi atribuirile tastelor de comenzi rapide.

## Meniul Insert (Inserare)




**Table** Permite crearea unui tabel nou cu ajutorul uneia din următoarele opțiuni: Table Wizard, Datasheet view, Design view, sau Import/Link Table.

**Query** Creează o nouă interogare cu ajutorul asistentului Wizard sau prin intermediul modului proiectare – Design view.

**Form** Permite crearea unui forumular nou cu ajutorul asistenților Wizard sau direct prin intermediul modului proiectare Design View.

**Report** Permite crearea unui raport nou în cadrul bazei de date cu ajutorul asistenților Wizard sau direct prin intermediul modului de proiectare Design View.

**Macro** Permite crearea unui nou macro..

**Module** Permite crearea unui modul nou şi afişarea secţiunii de declaraţii a acestuia într-o fereastră de modul.

**Class Module** Inserează o clasă de module care nu este asociată cu un formular sau raport al bazei de date, deschizând secţiunea de declaraţii a acesteia într-o fereastră de modul.

**AutoForm** Permite crearea automată a unui formular pornind de la tabelul sau interogarea selectată.

**AutoReport** Permite crearea automată a unui raport pornind de la tabelul sau interogarea selectată.

**Procedure** Apare în mod automat atunci când este deschisă o fereastră de modul şi permite inserarea unei proceduri la poziţia curentă a cursorului.

**Date/Time** Permite inserarea unei casete de text care va conţine o formulă pentru afişarea datei şi orei curente în paginile formularului sau raportului selectat.

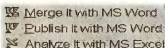
## Meniul Tools (Unelte)




**Spelling** Apelează corectorul gramatical (al limbii engleze) pentru verificarea din punct de verificare gramatical al conţinutului înscris în tabelele bazei de date sau în casetele de text ale unui formular.

**AutoCorrect** Personalizează setările corectorului gramatical pentru verificarea textului introdus de utilizator, modificând lista cuvintelor acestuia.

**OfficeLinks** ► permite exportul rezultatelor prelucrate de un formular sau raport al bazei de date în diferite formate recunoscute de aplicaţia Access.




►**Merge It with MS Word**

Lansează aplicația Microsoft Word Mail Merge Wizard, permițând astfel exportul datelor într-un fişier .doc.

►**Publish It with MS Word** Creează un fişier .rtf pornind de la un obiect selectat (Exemplu: tabel, raport).

►**Analyze It with MS Excel** Creează un fişier .xls pornind de la obiectul selectat. Implicit numele fişierului .xls este numele obiectului sursă.

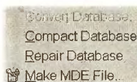
**Relationships** Afişează fereastra de relații astfel încât puteți creea, modifica sau edita relațiile dintre tabelele sau interogările bazei de date.

**Analyze** ►

► **Table** Lansează asistentul Table Analyzer Wizard, care permite fragmentarea tabelelor bazei de date în mai multe tabele (relaționate) cu o structură simplă pentru mărirea eficacității şi vitezei de prelucrare a datelor.

► **Performance** Lansează asistentul Performance Analyzer care analizează performanțele bazei de date şi obiectelor incluse în aceasta.

► **Documenter** Creează un raport asupra modului de funcționare şi proprietăților obiectelor bazei de date.




**Database Utilities** ►

►**Convert Database** Converteşte o bază de date dintr-o versiune mai veche în versiunea Microsoft Access 8.0.

►**Compact Database** Rearanjează într-un sistem continuu diferitele componente ale bazei de date, permițând astfel o mai rapidă accesare a datelor şi o îmbunătățire a performanțelor bazei de date.

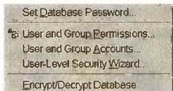
►**Repair Database** Reface greşelile care ar putea apare în structura de relații a bazei de date printr-o eventuală întrerupere forțată a alimentării unității sau printr-o funcționare defectuoasă a diferitelor componente hardware.

►**Make MDE File** Creează o bază de date compactată, compilată şi fără cod VBA pe care utilizatorul ar putea să-l modifice voit sau din greşeală.

**Security ►**

►**Set Database Password**

Permite modificarea parolei specifice unei baze de date. Numai utilizatorii cu drepturi de administrare a bazei de date şi cei care cunosc parola anterioară pot solicita modificarea parolei.




►**User And Group Permissions** Permite modificarea utilizatorilor şi grupurilor de lucru precum şi a drepturilor de acces ale acestora la diferitele obiecte ale bazei de date.

►**User And Group Accounts** Permite afişarea, tipărirea, adăugarea, ştergerea sau modificarea utilizatorilor sau grupurilor de lucru. De asemenea permite modificarea parolei administratorului implicit Admin.

►**User-Level Security Wizard** Lansează în execuţia asistentul Level Security Wizard cu ajutorul căruia poate fi creată o bază de date securizată împotriva utilizatorilor neautorizaţi.

►**Encrypt/Decrypt Database** Criptează sau decriptează baza de date. Criptarea face ca procesele de la nivelul bazei de date să se desfăşoare mai lent, însă în acest fel diferitele utilitare pentru editare nu vor avea putea interpreta nici măcar parţial informaţiile conţinute într-o astfel de bază de date.

**Replication ►**

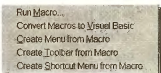
►**Synchronize Now** Sincronizează conţinutul a două baze de date.

►**Create Replica** Creează o copie replică a bazei de date curente. Copia poate fi creată şi prin “agăţarea şi aruncarea “ bazei de date curente în utilitarul My Briefcase de pe desktop.

►**Resolve Conflicts** Lansează în execuţie asistentul Conflict Resolver care permite identificarea şi eliminarea conflictelor care ar fi putut apare în urma unor sincronizări anterioare.

►**Startup** Permite personalizarea proprietăților de start-up ale bazei de date (formularul care va fi deschis în mod automat la lansarea în execuție, iconițele, titlul aplicației)

**Macro ►**




►**Run Macro** Lansează în execuție un macro selectat.

►**Convert Form's/Report's Macros To Visual Basic** Convertește un macro în instrucțiuni VBA.

►**Create Menu from Macro** Creează o bară de meniu pornind de la un macro selectat.

►**Create Toolbar from Macro** Creează o bară de unelte pornind de la un macro selectat.

►**Create Shortcut Menu from Macro** Creează un meniu scurtătură dintr-un macro selectat.

►**ActiveX Controls** Permite înregistrarea sau neânregistrarea controalelor ActiveX în baza de date curentă.

## Meniul Windows

**Cascade** Aranjează ferestrele MDI copil deschise, în cascadă, asemănător cărților de joc dintr-un pachet.

**Tile** Aranjează ferestrele MDI copil deschise, astfel încât să fie vizibile simultan și să aibă aceeași mărime.

## Meniul Help (Ajutor)

**Contents** Afișează o listă a subiectelor de asistență soft disponibile.

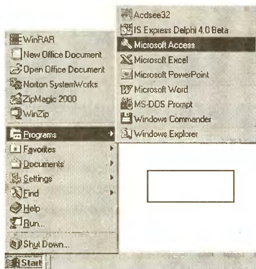
**Microsoft on the Web** Vă puteți conecta la locații Microsoft Web direct din Access utilizând comanda Microsoft pe Web (Microsoft on the Web) din meniul Ajutor (Help).

# NOŢIUNI GENERALE

## 1.1. Lansarea în execuţie

**Pasul 1:** Executaţi click cu mouse-ul pe **Start–Programs–Microsoft Access**

*1. În cazul în care meniul dumneavoastră de Start are o altă configuraţie decât cel din figura alăturată, deşi dumneavoastră aţi instalat corect aplicaţia Access 97, nu disperaţi. Căutaţi pe hard disk fişierul MSACCESS.EXE" şi lansaţi-l în execuţie.*




**Pasul 1:** Încercaţi să creaţi o scurtătură către fişierul "Msaccess.exe" şi introduceţi în linia de comandă (câmpul Target) a acesteia următoarea secvenţă: "C:\Program Files\Microsoft Office\Office\MSACCESS.EXE " /User DSC după care lansaţi-o în execuţie.

**Pasul 2:** În caseta de dialog care apare sunteţi invitat să precizaţi numele sau tipul bazei de date pe care doriţi să o deschideţi sau să o creaţi.

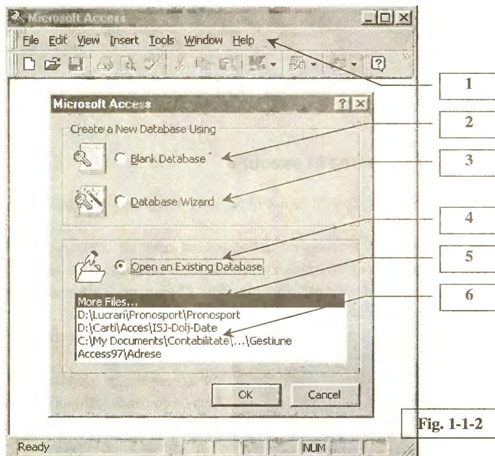


Fig. 1-1-2

Puteți astfel să optați fie pentru crearea unei baze de date noi (executând un click pe una din opțiunile notate cu 2 – pentru crearea unei baze de date fără înregistrări – sau 3 – pentru a activa asistentul Wizard ce vă va ajuta să creați o bază de date de un anumit tip), fie pentru deschiderea unei baze de date (4). Caseta de dialog afişează calea şi numele ultimelor patru baze de date deschise de utilizator (6). În cazul în care baza de date pe care doriți s-o deschideți nu se află în această listă programul vă oferă opțiunea "More Files …" (5). Precizarea acțiunii dorite (creare de baze de date noi sau deschiderea unei baze de date existente) se face prin marcarea (prin executarea unui click pe numele opțiunii) acelei opțiuni şi executarea

unui click pe butonul "OK" (sau pe butonul "Cancel" pentru a renunța la ajutorul oferit de această casetă de dialog.

În figura de mai sus s-a optat pentru deschiderea unei alte baze de date. De menționat că bara de meniu (1) nu poate fi apelată atâta timp cât caseta de dialog este vizibilă.

Fig. 1-1-3

**Pasul 3:** În caseta de dialog care apare sunteți invitat să precizați numele sau tipul bazei de date pe care doriți să o deschideți cu ajutorul următoarelor controale:

1. **Lista derulantă "File Name:"** - Numele bazei de date. Introduceți numele bazei de date pe care doriți să o găsiți.
2. **Lista derulantă "Files of Type:"** - Tipul fişierului bază de date.
4. **Lista derulantă "Look in:"** - Afişează fişierele sau folderele din unitatea sau folderul selectat în caseta **Caută în:**

Faceți click pe unitatea sau folderul ce conține fişierul dorit. Conținutul unității sau al folderului selectat de dvs. va apare dedesubt în lista cu foldere (**3.**) împreună cu fişierele corespunzătoare tipului selectat de dumneavoastră.

5. **Butonul "Up One Level"** - Deschide folderul care este imediat superior ca nivel celui curent.

6. **Lista derulantă “Last modified”** - Vă permite căutarea definirea perioadei de timp în care fişierele căutate au fost modificate ultima oară.
7. **Lista derulantă “Text or Property”** - Vă permite să precizați textul pe care doriți să-l găsiți. Pentru a găsi expresii, puneți textul între ghilimele (" "). De exemplu, pentru a găsi numai fişierele care conțin expresia dans modern, tastați "dans modern".
8. **Butonul “Find Now”**- Afişează fişierele care întrunesc criteriile de căutare precizate de dvs. în stânga. Pe parcursul căutării, numele butonului se schimbă în Stop. Pentru a abandona căutarea, faceți clic pe butonul Stop.
9. **Caseta “Exclusive”**- Validați (prin bifare!) această casetă dacă doriți ca baza de date să nu mai poată fi deschisă de alți utilizatori pe perioada în care lucrați dumneavoastră.
10. **Butonul “Open”**- Aplică şi salvează orice modificare făcută de dvs. şi închide această casetă de dialog.

## 1.2. Fereastra Database

În Access 97 veți lucra în special cu această fereastră, care vă va ajuta să selectați, creați sau să modificați obiectele necesare funcționării unei baze de date. Această fereastră acceptă şase tipuri de obiecte:



Fig. 1-2-1

**Tables** – Tabele . Sunt "foile de calcul tabelar" ale unei baze de date. La deschiderea unui tabel datele sunt afişate pe ecran organizate pe linii şi pe coloane. În cadrul unei baze de date tabelele stochează datele ca înregistrări.

**Query** – Interogări. Sunt cererile adresate tabelelor pentru extragerea datelor şi afişarea organizată a acestora într-un anumit mod.

**Forms** – Formulare. Oferă o interfaţă atractivă pentru vizualizarea sau introducerea datelor din tabele. Prin faptul că urmăresc introducerea sau modificarea datelor din tabele după anumite reguli stricte , impuse de programator, permit eliminarea erorilor provocate de utilizatori la manipularea datelor.

**Reports** – Rapoarte. Permit extragerea datelor din tabele (eventual cu ajutorul interogărilor) şi de a le prezenta în forma în care acestea vor fi tipărite pe hârtie.

**Macros** – Macroinstrucţiuni. Conţin colecţii de comenzi – specifice Access 97 – care vor fi executate pentru realizarea unui anumit obiectiv. Sunt folosite de regulă pentru crearea de meniuri personalizate în cadrul unei aplicaţii şi pentru executarea unor comenzi simple.

**Module** – Module. Conţin instrucţiuni VBA (Visual Basic for Applications – limbajul de programare folosit de aplicaţiile Microsoft Office), grupate în "subrutine" sau "funcţii". Despre subrutine şi funcţii vom discuta în ultimul capitol al acestei lucrări.

Pentru a crea obiecte noi în cadrul unei baze de date, după ce executaţi click pe eticheta tipului de obiect, executaţi un click pe butonul "New" (sau apăsaţi combinaţia de taste [ALT]+[N].

Pentru a modifica sau pentru a deschide un obiect oarecare din baza dumneavoastră de date, după ce selectaţi acel obiect (prin executarea unui click pe numele acestuia) executaţi un click pe butonul "Design" (pentru modificări!) sau pe butonul "Open" (pentru a-l deschide!).

Atunci când nu aveţi selectat nici un obiect din listă (şi acest lucru îl puteţi simula foarte simplu executând un click oriunde în zona liberă a listei

de obiecte) observați că butoanele “Open” și “Design” sunt “stinse” (expresiile Open și Design de pe butoane își pierd strălucirea), ceea ce ar trebui să însemne pentru dumneavoastră că operațiunile de deschidere și modificare ale unui obiect sunt în acel moment indisponibile.

## 1.3. Tabele (Table)

Tabelele (Tables) sunt foi de calcul tabelar ale unei baze de date, care organizează pe linii și coloane datele afișate pe ecran. În cadrul bazei de date tabelele stochează datele ca înregistrări, care corespund în general cu liniile din modul de afișare “Table”.

Access 97 permite încorporarea tabelelor în cadrul bazei de date ale unei aplicații precum și preluarea datelor din tabelele ale unei alte aplicații – prin crearea unui tabel fals , numit tabel atașat.

Spre deosebire de tabelele unei baze de date, tabelele atașate permit importul informațiilor dintr-o bază de date externă (Microsoft Access versiune 1.x, 2.0, 7.0/95, sau 8.0/97), sau din fișiere de date de format diferit cum ar fi: Microsoft Excel, dBASE, Microsoft FoxPro, sau Paradox.

### 1.3.1. Tipuri de date

În interiorul unui tabel datele sunt organizate pe linii și pe coloane pentru a ușura citirea acestora de către utilizatori. Intersecția unei linii cu o coloană (locul în care se introduc efectiv datele) o vom numi în continuare “câmp”. Pentru ca un câmp oarecare dintr-un tabel să accepte un anumit tip de date introdus de utilizator este necesar ca acelui câmp să-i fie atribuite una din următoarele caracteristici:

| | |
|---|---|
| **Text** | Un set de maximum 255 caractere (inclusiv semnele de punctuație și cifrele). Implicit Access 97 propune ca lungimea acestui câmp să fie de 50 caractere – ceea ce în majoritatea cazurilor este suficient. |
| **Memo** | Asemănător cu tipul Text cu deosebirea că lungimea setului de caractere este practic nelimitată. |
| **Number** | Un număr întreg sau în virgulă mobilă. Sunt admise șase subtipuri. |
| **Date/Time** | O dată calendaristică sau o oră în unul din cele șapte subtipuri posibile. |
| **Currency** | Un număr formatat cu două zecimale, care folosește virgula ca separator între ordinele de mărime și parantezele pentru încadrarea valorilor negative. Admite șase subtipuri. |
| **Autonumber** | Un număr întreg care este incrementat în mod automat pe măsură ce sunt introduse noi înregistrări în tabel. Sunt admise toate cele șase subtipuri ale datelor de tip numeric (Number). Recomandăm însă folosirea subtipului Long Integer. |
| **Yes/No** | Admite date de tip logic (boolean) pentru a stoca valori simple (Adevărat/Fals = True/False, Activ/Inactiv = On/Off, Da/Nu = Yes/No) |
| **OLE Object** | Un alt tip de date cum ar fi imagini, sunete, etc. |
| **Hyperlink** | O referință la un document extern, care, atunci când este selectată, deschide documentul respectiv în modul de afișare corespunzător. |

### 1.3.2. Subtipuri de date

Tipul de date **Number** admite şase subtipuri de date:

| | |
|---|---|
| **Byte** | Valori întregi de la 0 la 255. Este folosit atunci când spaţiul pe disk reprezintă o problemă pentru utilizator. Fiecare valoare introdusă ocupă un octet de memorie. |
| **Integer** | Numere întregi de la –32.768 la +32.767. Fiecare valoare introdusă ocupă doi octeți. |
| **Long Integer** | Numere întregi cuprinse între -2,147,483,648 şi +2,147,483,647. Fiecare valoare introdusă ocupă 4 octeți. |
| **Single** | Aproape toate numerele în virgulă mobilă cu până la şapte zecimale, pe patru octeţi. |
| **Double** | Numere în virgulă mobilă cu până la 15 zecimale, pe opt octeți. Folosit pentru calcule care solicită o precizie ridicată. |
| **Replication ID** | Un număr special, folosit intern de Access 97 pentru a identifica în mod unic obiectele unei baze de date. Fiecare valoare ocupă 16 octeți. |

Tipul de date **Date/Time** admite şapte subtipuri de date:

| | |
|---|---|
| **General Date** | Stochează data şi ora. Acest subtip de date este o combinaţie între subtipurile **Short Date** şi **Long Date** şi depinde de modul în care aceste subtipuri au fost definite în caseta de dialog **Regional Settings Properties** a sistemului de operare Windows. Exemplu: 04/12/1993 05:34:00 PM |
| **Short Date** | Asemănător cu modul în care a fost definit în caseta de dialog **Regional Settings Properties** a sistemului de operare Windows. Exemplu: 04/12/93 |
| **Long Date** | Asemănător cu modul în care a fost definit în caseta de dialog **Regionale Settings Properties** a sistemului de operare Windows. Exemplu: Saturday, April 3, 1993. |

| | |
|---|---|
| **Medium Date** | Exemplu: 3-Apr-93. |
| **Long Time** | Asemănător cu modul în care a fost definit în caseta de dialog **Regional Settings Properties** a sistemului de operare Windows. Exemplu: 5:34:23 PM. |
| **Medium Time** | Exemplu: 5:34 PM. |
| **Short time** | Exemplu: 17:34. |

Tipul de date **Currency** admite şase subtipuri de date:

| | |
|---|---|
| **General Number** | Afişează numerele în forma în care au fost introduse. |
| **Currency** | Foloseşte virgula ca separator între ordinele de mărime şi afişează cel puţin două zecimale. |
| **Fixed** | Afişează cel puţin o cifră a unităţilor şi un număr definit de zecimale. |
| **Standard** | Foloseşte virgula ca separator al ordinelor de mărime şi cel puţin două zecimale. |
| **Percent** | Înmulţeşte valoarea introdusă cu 100 şi adaugă semnul %. |
| **Scientific** | Foloseşte notaţia ştiinţifică pentru exprimarea numerelor. Exemplu: 3.46E+3 (346.000) |

### 1.3.3. Crearea unei baze de date

În esenţă crearea unei baze de date cu ajutorul programului Access 97 se rezumă la crearea unui fişier cu extensia .mdb chiar de către program.

**Pasul 1:** Pentru aceasta din meniul **File** al programului executaţi un click pe opţiunea **New Database...**. Dacă parcurs acest pas atunci ecranul calculatorului dumneavoastră va afişa imaginea din fig. 1-3-3-1:

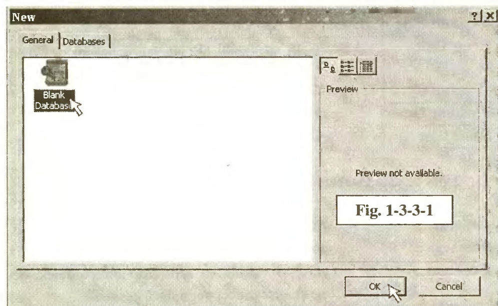


Fig. 1-3-3-1

**Pasul 2:** Executați în continuare un click pe iconița “Blank Database” (Bază de date fără înregistrări) iar apoi un click pe butonul **OK**.

Pe ecranul calculatorului dumneavoastră va apare o casetă de dialog asemănătoare cu cea din figura 1-3-3-2:

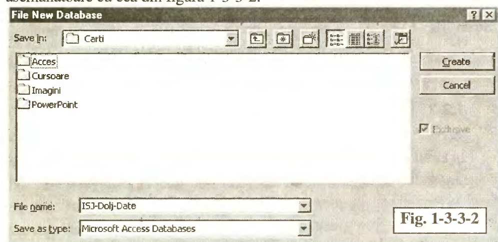


Fig. 1-3-3-2

Această casetă “File New Database” este în esență identică în privința funcționarii cu caseta de dialog din figura 1-1-3, diferind de aceasta prin

etichetele atribuite diverselor controale şi prin numărul mai mic de funcţii pe care le pune la dispoziţia dumneavoastră.

Nu uitaţi să precizaţi că doriţi crearea unei baze de date de tip Access 97 selectând opţiunea "Microsoft Access Databases" din câmpul "Save as type:"

**Pasul 3:** Dacă toţi aceşti paşi au fost parcurşi ceea ce vă mai rămâne de făcut este să executaţi un click pe butonul "Create".

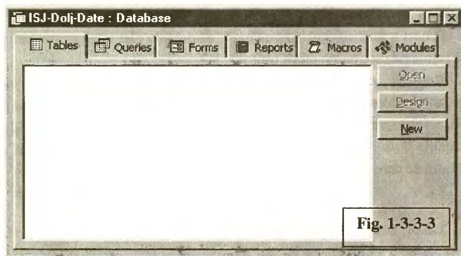


Fig. 1-3-3-3

În urma acestei acţiuni din partea dumneavoastră, în fereastra "Microsoft Access" a programului Access 97 va apare această fereastra cu titlul "ISJ-Dolj-Date".

⊶Încercaţi să creaţi o bază de date executând însă un click pe eticheta "Databases" (caseta de dialog New din figura 1-3-3-1), urmat de un dublu click pe una din iconiţele care vor apare în listă.

### 1.3.4. Deschiderea unei baze de date

Odată creată, o bază de date Access 97 (un fişier cu extensia .mdb) poate fi deschisă din Windows Explorer prin executarea unui dublu click pe numele sau pe iconiţa acesteia.

Dacă programul Access 97 a fost deja lansat în lucru, deschiderea unei baze de date se face prin apelarea opţiunii **Open Database ...** din meniul **File** Pentru aceasta executaţi un click pe meniul **File** apoi un click pe opţiunea **Open Database ....**

Pentru a finaliza deschiderea bazei de date de care aveţi nevoie reluaţi explicaţiile pe care le-am oferit figurii 1-1-3.

**Reţineţi**: deschiderea unei alte baze de date presupune în mod automat închiderea bazei de date curente (chiar dacă dumneavoastră nu doriţi acest lucru). Dacă doriţi sau aveţi nevoie să aveţi deschise simultan mai multe baze de date este necesar să lansaţi în execuţie de tot atâtea ori câte baze de date vreţi să aveţi deschise, programul Access 97.

### 1.3.5. Crearea tabelelor

Înainte de a începe crearea tabelelor unei baze de date este bine să:

1. Studiaţi cu atenţie datele pe care le veţi introduce şi să determinaţi formatul acestora.
2. Analizaţi datele pentru a stabili relaţiile dintre acestea.
3. Prelucraţi şi separaţi datele pentru a putea crea tabele cât mai mici şi cât mai uşor de înţeles de către cei care vor lucra cu ele.
4. Vă gândiţi cum puteţi folosi toate facilităţile programului Access 97.
5. Solicitaţi ajutorul unor colegi – este surprinzător cât de multe idei pentru structurarea unei baze de date vă pot furniza alţii.

Presupunând că tocmai ați deschis baza de date "ISJ-Dolj-Date" imaginea pe care o va afișa calculatorul dumneavoastră ar trebui să fie asemănătoare cu cea din figura 1-2-1. Dacă eticheta **Table** nu se află în prim plan executați un click pe aceasta pentru a preciza programului că intenționați să întreprindeți o acțiune asupra tabelelor acestei baze de date.

Executați în continuare un click pe butonul **New** pentru a trece la crearea primului tabel. În caseta de dialog care va apare (fig. 1-3-5-1) sunteți invitat să precizați modul în care veți crea tabelul.

Înțelegerea modurilor în care poate fi creat un tabel va fi tratată în capitolele imediat următoare. Recomandăm folosirea modului Design View pentru posibilitățile foarte mari pe care le pune la dispoziția utilizatorului.

Alegerea modului în care va fi creat un tabel se face prin executarea unui click pe numele modului de lucru (Datasheet View, Design View, etc.) urmat de un click pe butonul **OK**.

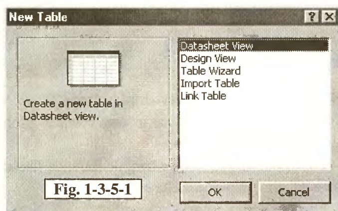



Fig. 1-3-5-1

În continuare crearea tabelelor va fi tratată de la acest punct în capitole-le 1-3-5-1 -> Modul Design View, 1-3-5-2 Modul Table Wizard, 1-3-5-3 -> Modul Import Table, 1-3-5-4 -> Modul Link Table.

### 1.3.5.1. Modul Design View (Modul Proiectare).

Dacă ați efectuat toți pașii de mai sus și ați deschis fereastra "New Table" din figura 1-3-5-1, executați un click pe opțiunea "Design View" din listă, apoi un click pe butonul "OK" (sau apăsați pe tasta ENTER).

Rezultatul acestei acțiuni va fi deschiderea unei ferestre cu numele generic "Table1", în care sunteți invitat să precizați:

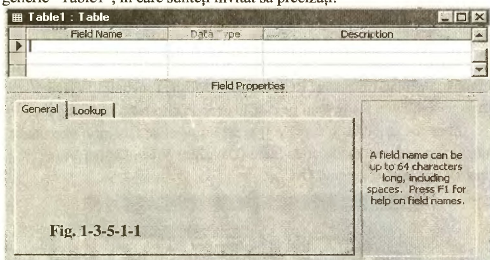



Fig. 1-3-5-1-1

1. Numele coloanelor tabelului dumneavoastră (coloana "***Field Name***").
2. Tipul datelor pe care le veți introduce în acea coloană. (Coloana "***Data Type***").
3. Descrierea (explicația) ce va apare pe bara de stare, de fiecare dată când veți intra într-unul din câmpurile acelei coloane.

Descrierea nu este importantă pentru programare, ea va constitui un ajutor de neprețuit când va trebui să vă descurcați în hățișul de date și informații ale unei baze de date.

Ne propunem să realizăm un tabel în care să introducem denumirea județelor. Pentru aceasta definiți structura tabelului dumneavoastră introducând de la tastatură ca în fig. 1-3-5-1-2.

Am definit astfel câmpurile:

1. ***IdJudet*** de tip "***AutoNumber***" – numărător automat – pe care îl vom folosi pentru identificarea unică a unui județ.
2. ***Prescurtare*** de tip "***Text***" – pe care-l vom folosi pentru a memora prescurtarea auto a județului.
3. ***NumeJudet*** de tip "***Text***" – pe care îl vom folosi pentru a memora denumirea județului.

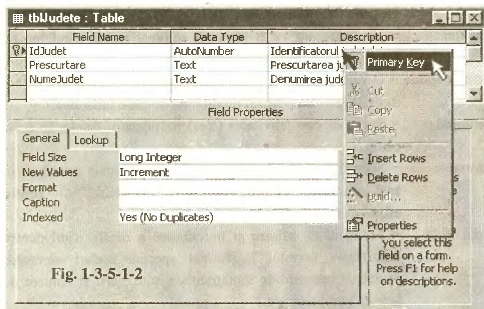



**Fig. 1-3-5-1-2**

Definiți, în continuare, un index unic pentru coloana IdJudet prin executarea unui click cu butonul drept al mouse-ului, oriunde în interiorul rândului IdJudet din fig. 1-3-5-1-2 , urmat de executarea unui click pe opțiunea Primary Key. Indexul nu va admite introducerea valorilor identice.

Pe lângă tipul de date care poate fi memorat de către un câmp, este necesar, uneori să precizăm şi alte caracteristici (proprietăți) ale câmpurilor, ca de exemplu:

Field Size = mărimea câmpului

New Values = valoarea care va fi atribuită în mod automat la introducerea noilor date într-un astfel de câmp. (Random – generează un

număr aleator de tip Long Integer – Increment – numără în mod automat înregistrările introduse în acel câmp.) Recomandăm proprietatea Increment.

Indexed = creează sau nu (No) un index pentru rapidizarea căutărilor ulterioare a datelor din acel câmp, care poate bloca introducerea din greşeală a valorilor identice de către utilizatori (Yes (No Duplicates)), sau poate bloca o astfel de greşeală (Yes (Duplicates OK)).

Caption = Eticheta care va fi preluată automat atunci când într-un formular va fi folosit acel câmp.

Default Value = O valoare care va fi introdusă automat la adăugarea de noi înregistrări. (Va putea fi modificată de către utilizator!).

Required = Specifică obligativitatea (Yes) introducerii datelor în acest câmp.

Format = Specifică modul în care vor fi afişate datele introduse în acel câmp.(> - pentru afişarea cu majuscule, < - pentru afişarea cu litere mici)

Input Mask = Formatul de afişare şi introducere a datelor (util pentru lizibilitatea textului în format special: coduri numerice personale, numere de înmatriculare auto, serii şi numere de acte de identitate, etc.)

Închideți fereastra Table1:Table, în care ați definit caracteristicile câmpurilor, executând un click pe butonul de închidere al ferestrei, pentru a salva tabelul creat.

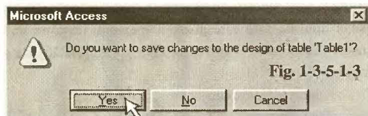



Fig. 1-3-5-1-3

Dacă în acest moment veți apăsa pe butonul ***Cancel***, veți reveni în modul Proiectare.

Apăsând însă pe butonul ***Yes*** veți primi caseta de dialog următoare care vă propune același nume generic Table1 pentru numele tabelului pe care l-ați creat. Modificați și atribuiți tabelului dumneavoastră numele "tblJudete".

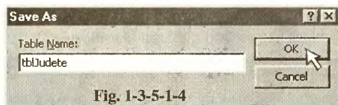


Fig. 1-3-5-1-4

Încercați să creați un alt tabel cu numele "tblLocalitati" cu următoarele proprietăți:

| | |
|---|---|
| IdLocalitate | AutoNumber, Increment, Primary Key |
| Localitate | Text, 50, Neindexat |
| Comuna | Text, 50, Neindexat |
| CodLocalitate | Text, 5, Neindexat |
| IdJudet | Number, Long Integer, Indexat (Duplicates OK) |

### 1.3.5.2. Modul Table Wizard – Crearea tabelelor cu ajutorul asistenților Wizard.

Asistenții Wizard vă oferă numeroase elemente de automatizare a operațiunilor pe care le întreprindeți. Vă va trebui însă foarte mult timp pentru a vă familiariza cu toate facilitățile oferite de aceștia. Este indicată însă studierea acestora.

În exemplul următor vom crea cu ajutorul unui asistent Wizard un tabel cu numele AdresaElevi. Pentru aceasta este necesar să obțineți pe ecran caseta de dialog din fig. 1-3-5-1 și după ce veți executa un click pe opțiunea "Table Wizard" din listă, "apăsați" pe butonul OK.

În acest moment asistentul Wizard pentru crearea tabelelor vă propune o serie de tabele grupate în două categorii: Business – Afaceri şi Personal. Dacă veţi bifa opţiunea Personal, în lista "Sample Tables" va fi afişată o listă cu modele de tabele considerate de uz personal. Executaţi un click pe modelul "Addresses", în cazul în care acesta nu este deja marcat aşa cum apare în fig. 1-3-5-2-1. În lista "Sample Fields" vi se va propune imediat o listă de tipuri de coloane ce pot fi folosite de acest tabel – pe care nu sunteţi obligat să o preluaţi şi folosiţi în totalitate.

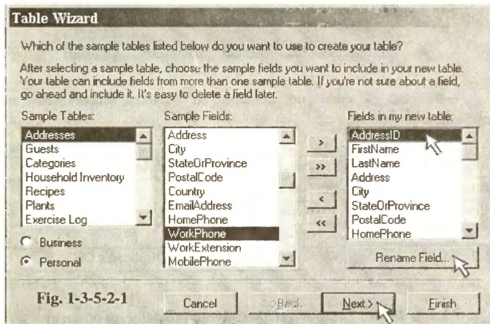


Fig. 1-3-5-2-1

Pentru a prelua doar coloanele de care aveţi nevoie selectaţi-le (prin executarea unui click pe nume) şi apăsaţi pe butonul marcat cu ">". În lista "Fields in my table" vor apare numele coloanelor preluate de dumneavoastră. Puteţi oricând să eliminaţi oricare din coloanele preluate executând un click pe numele său din lista "Fields in my table", urmat de un click pe butonul marcat cu semnul "<". Pentru a schimba numele unei coloane din lista celor preluate, selectaţi-o şi apăsaţi pe butonul "Rename Field". – Nu uitaţi să transferaţi mai întâi toate coloanele, aşa cum apar în

fig. 1-3-5-2-1, pentru a putea realiza tabelul de care vom avea nevoie pentru aplicația noastră.

Redenumiți coloanele preluate așa cum apar în figura următoare.

Dacă ați redenumit ca în figura de mai sus coloanele tabelului dumneavoastră, apăsați pe butonul "Next" pentru a trece la pasul următor.

**Rețineți**: puteți întrerupe și renunța oricând la crearea unui tabel cu ajutorul asistenților Wizard apăsând tasta Escape sau executând un click pe butonul Cancel. De asemenea puteți finaliza tabelul creat pe această cale apăsând pe butonul Finish – dar numai în condițiile unei cunoașteri foarte bune a pașilor urmați de acești asistenți.

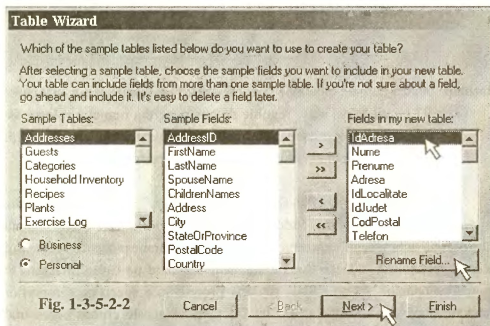



Fig. 1-3-5-2-2

În pasul următor sunteți invitat să precizați numele tabelului în câmpul marcat cu cifra 1 în figură. Denumiți-l "AdreseElevi"!

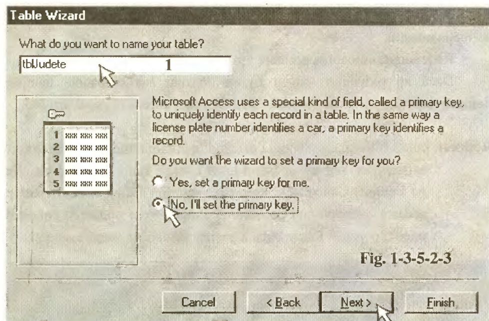


Fig. 1-3-5-2-3

**Rețineți**: pentru numele tabelelor şi al obiectelor pe care le creați în Access 97 puteți folosi toate regulile de compunere a numelui folosite de Windows. Recomandăm omiterea spațiilor din numele obiectelor pentru a preveni eventualele confuzii din partea programului.

Bifați opțiunea "***No, I'll set the primary key.***" pentru a nu permite asistentului atribuirea unui index unic după regulile sale deoarece nu întotdeauna aceste reguli corespund necesităților dumneavoastră. Asistentul Wizard recunoaşte indecşii după denumirile standard pe care utilizatorii le acordă coloanelor. (ID; CODE) şi după proprietățile specifice câmpurilor indexate după o cheie unică, dar nu întotdeauna aceste criterii corespund necesităților dumneavoastră. Indiferent de decizia pe care o veți lua în acest moment , nu uitați că oricând veți putea modifica structura unui tabel deschizându-l în modul proiectare (Design View).

Pentru a trece la pasul următor apăsați pe butonul Next.

În figura 1-3-5-2-4 veți regăsi numele coloanelor definite de dumneavoastră, deschizând caseta "combo" aşa cum se arată la poz. 1.

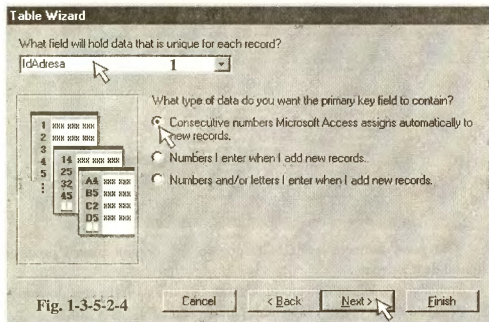


Fig. 1-3-5-2-4

Alegeți "IdAdresa" pentru indexul unic. În continuare va trebui să precizați modalitatea în care va fi realizată numerotarea înregistrărilor în tabelul creat:

1. În mod automat atunci când veți adăuga o nouă înregistrare ("Consecutive numbers Microsoft Access ...).

2. Manual de către utilizator la introducerea unei noi înregistrări. ("Numbers I enter when I add new records")

3. Manual, prin introducerea unui număr unic sau a unei combinații unice de caractere de către utilizator. ("Numbers and/or letters ...")

Alegeți prima opțiune, ca în figura 1-3-5-2-4 și apăsați pe butonul "Next" pentru a trece la pasul următor.

Dacă v-ați gândit să stabiliți o relație între datele din tabelul dumneavoastră "AdreseElevi" și un alt tabel (presupunem "tblLocalitati"), efectuați pașii din fig. 1-3-5-2-5 fără să apăsați pe butonul Next. Stabilirea unei relații între cele două tabele va fi evidențiată în fig. 1-3-5-2-6.

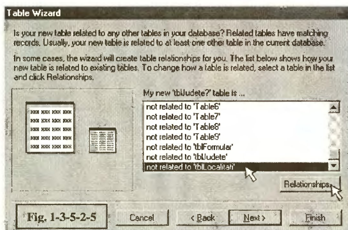


Fig. 1-3-5-2-5

În acest moment aveți la dispoziție trei opțiuni pentru precizarea tipului de relație:

1. Anularea oricărei relații sau precizarea faptului că între cele două tabele nu există nici o relație ("The tables aren't related").
2. O înregistrare din tabelul "AdreseElevi" are mai multe corresponden-țe în tabelul "tblLocalitati". – One record in the 'AdreseElevi' table ...
3. O înregistrare din tabelul "tblLocalitati" are mai multe corresponden-țe în tabelul "AdreseElevi". – One record in the 'tblLocalitati' table ...

Bifați ultima opțiune și executați un click pe butonul OK.

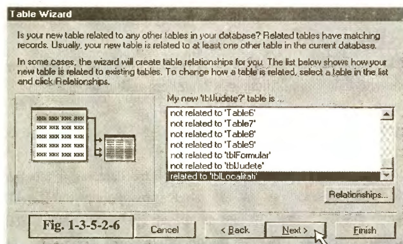


Fig. 1-3-5-2-6

Observați modalitatea în care Access 97 vă atrage atenția asupra tipului de relație pe care a realizat-o conform precizărilor dumneavoastră.

Apăsați pe butonul Next pentru a trece la pasul următor.

În continuare asistentul Wizard vă propune alegerea uneia din cele trei acțiuni pe care le poate efectua după ce veți apăsa pe butonul "Finish" și după ce va crea tabelul dumneavoastră:

1. Reintrarea în modul de lucru proiectare pentru a putea modifica manual structura tabelului pe care l-a creat pentru dumneavoastră.

2. Deschiderea tabelului pentru introducerea manuală a datelor "Enter data directly into the table."

3. Crearea unui formular pentru introducerea datelor. "Enter data into the table using a form…".

Optați pentru cea de-a doua variantă propusă și apăsați pe butonul Finish așa cum este indicat în fig. 1-3-5-2-7.

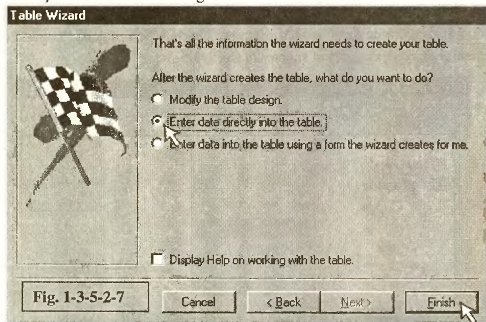


Fig. 1-3-5-2-7

Tabelul dumneavoastră a fost creat în acest moment și tot ceea ce vă mai rămâne de făcut este să introduceți datele. În final închideți acest tabel pentru a reveni la fereastra bazei dumneavoastră de date.

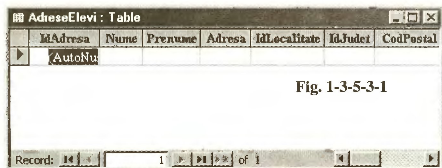


Fig. 1-3-5-3-1

### 1.3.5.3. Modul Import Table

Dacă în alte baze de date Access 97 există tabele a căror structură sau conținut poate fi considerată utilă aplicației dumneavoastră nu ezitați s-o preluați pentru a evita efortul creării sale. Pentru aceasta executați un click cu butonul drept al mouse-ului în interiorul listei cu tabelele bazei dumneavoastră de date iar apoi executați un click pe opțiunea "Import" din meniul de context așa cum sugerează fig. 1-3-5-3-1.

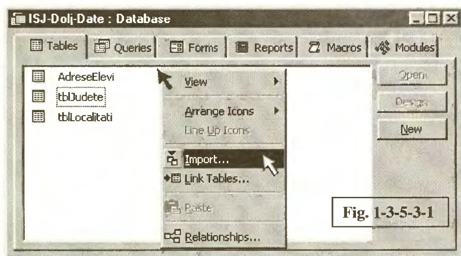


Fig. 1-3-5-3-1

Pentru a înțelege funcționarea ferestrei din revedeți explicațiile de la fig. 1-1-3. După selectarea bazei de date din care se va face importul unui tabel apăsați pe butonul "Import".

Conform fig. 1-3-5-3-3 puteți selecta chiar mai multe tabele (prin click-uri succesive) pe care să le introduceți în baza de date. Deselecția unui obiect selectat din greșeală se face prin executarea a încă unui click pe numele obiectului selectat. Apăsați pe butonul OK după selecția obiectelor care vor fi importate.

În urma acestei operații tabelele vor fi preluate și introduse în baza dumneavoastră de date.

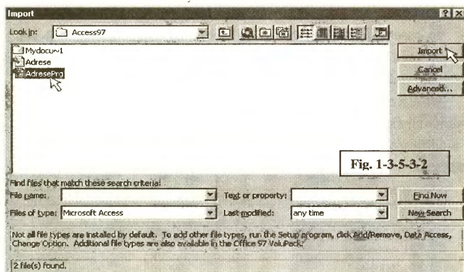


Fig. 1-3-5-3-2

### 1.3.5.4. Modul Link Table

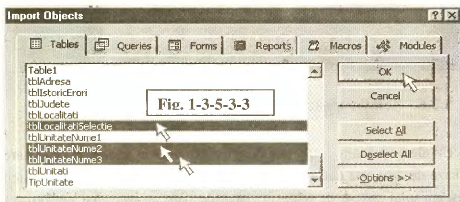


Fig. 1-3-5-3-3

Preluarea datelor din alte fişiere sau din tabelele unei alte baze de date se poate face nu numai prin importarea acelor fişiere ci şi prin realizarea unei "scurtături" la fişierele sau tabelele unei alte baze de date. Avantajul acestei metode este reducerea mărimii bazei de date prin neincluderea înregistrărilor în fişierul Access. Pentru realizarea ataşării unui tabel sau fişier extern la baza dumneavoastră de date, executați un click cu butonul drept al mouse-ului în interiorul listei tabelelor bazei de date şi din meniul de context executați un click pe opțiunea Link. (v. fig. 1-3-5-3-1).

În caseta de dialog "Link" va trebui, la fel ca în fig. 1-3-5-3-2 să selectați baza de date care conține tabelele sau obiectele pe care doriți să le ataşați bazei dumneavoastră de date.

Observați tipul de iconiță pe care aplicația Access 97 o atribuie în mod automat tabelelor ataşate. De reținut că proprietățile unui tabel ataşat nu pot fi modificate de către Access 97 ci doar datele pot fi prelucrate şi modificate.




### 1.3.6. Relaționarea tabelelor

Un element foarte important care trebuie avut în vedere la proiectarea unei baze de date este definirea sistemului de relații dintre tabelele acesteia. În absența unui sistem de relații, o bază de date nu este mai performantă

decât o foaie de calcul tabelar, irosindu-se astfel toate posibilitățile oferite de mecanismul bazei de date. Este dificil de conceput o bază de date utilă care să nu includă câteva relații între tabele.

Tipurile de relații pe care le puteți defini între tabelele unei baze de date sunt:

1. One To One – Unu la unu. Aceasta înseamnă că o înregistrare dintr-un tabel are corespondent doar o singură înregistrare dintr-un alt tabel.

2. One To Many – Unu la mai mulți. O înregistrare din tabelul sursă are mai multe corespondențe în celălalt tabel.

3. Many To One – Mai mulți la unul. Mai multe înregistrări dintr-un tabel au corespondent o singură înregistrare într-un alt tabel.

Pentru a crea relații între tabele procedați astfel:

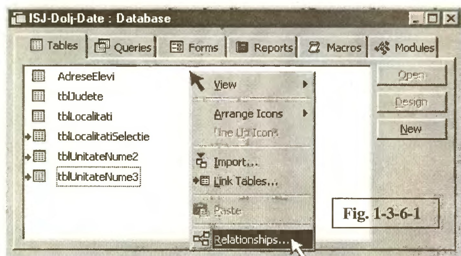


Fig. 1-3-6-1

**Pasul 1:** Executați un click cu butonul drept al mouse-ului în fereastra bazei de date și din meniul de context executați un click pe opțiunea "Relationships".

**Pasul 2:** Dacă este prima când deschideți fereastra "Relationships" a bazei de date, va fi afișată și fereastra "Show Table", care vă permite să stabiliți noi relații (sau să modificați relațiile existente) între tabelele și interogările dumneavoastră (fig. 1-3-6-2). Pentru a adăuga un tabel sau o interogare în fereastra Relationships,

executați un click pe numele unui tabel (sau interogări) din fereastra Show Table iar apoi un click pe butonul OK. Reluați operația până la introducerea tuturor tabelelor în fereastra Relationships. În final apăsați pe butonul Close pentru a închide caseta Show Table.

**Pasul 3:** Pentru a stabili acum o relație între tabelul tblJudete și tabelul tblLocalitati, prin intermediul câmpului IdJudete, agățați câmpul IdJudet din tblJudet și aruncați-l peste IdJudet din tblLocalitati, așa cum se arată în fig. 1-3-6-3.

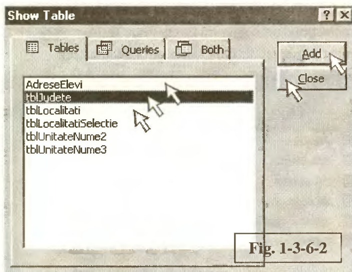


Fig. 1-3-6-2

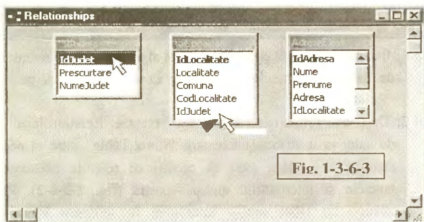


Fig. 1-3-6-3

**Rețineți:** pentru stabilirea unei relații între două tabele, nu prezența unui câmp cu nume identic este necesară, ci prezența a două câmpuri care conțin același tip de date. În exemplul nostru IdJudet din tblJudete este de tip AutoNumber - Long Integer iar IdJudet din tblLocalitati este de tip Number – Long Integer. Este recomandat să asigurați, înainte de a începe crearea unei relații, indexarea după o cheie unică (Yes (No Duplicates)) a câmpului din tabelul "Master" (în situația de față tblJudete este tabelul "Master" iar tblLocalitati este tabelul "Slave" – care este indexat, dar nu după o cheie unică). Nerespectarea acestor indicații va duce la eșuarea stabilirii relației dorite de dumneavoastră.

În definirea sistemului de relații există un sistem bine definit. Vă veți da seama că majoritatea bazelor de date bine proiectate folosesc mai multe tabele de dimensiuni mici pentru a înregistra elemente care trebuie să fie referite de alte tabele.

**Pasul 4:** În cazul în care agățarea și aruncarea unui câmp nu a fost executată corect, aveți posibilitatea restabilirii relației dintre două câmpuri, prin modificarea numelui câmpurilor înscrise în listele "Table/Query" și "Related Table/Query".

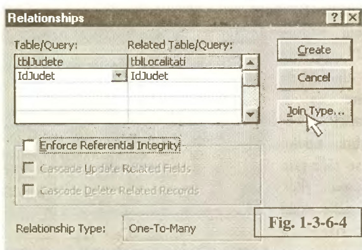


Fig. 1-3-6-4

Bifarea casetei de validare "Enforce Referential Integrity", are ca efect activarea şi aplicarea integrităţii referenţiale la nivelul câmpurilor alese.

Integritatea referenţială este un sistem de reguli Microsoft Access, folosit pentru asigurarea relaţionării câmpurilor valide, care nu permite ştergerea sau modificarea accidentală a datelor relaţionate. Integritatea referenţială poate fi activată atunci când sunt cunoscute şi respectate următoarele reguli:

Câmpul relaţionat din tabelul master este o cheie primară sau are un index unic.

Câmpurile relaţionate conţin acelaşi tip de date. Sunt admise totuşi două excepţii:

1. Un câmp de tip Autonumber poate fi relaţionat cu un câmp de tip Number dacă acesta conţine valori de tip Long Integer.

2. Un câmp de tip Autonumber care conţine date de tip Replication ID poate fi relaţionat cu un câmp de tip Number dacă acesta conţine date de acelaşi tip (Replication ID).

Ambele tabele aparţin aceleiaşi baze de date Microsoft Access.

Dacă tabelele care se relaţionează sunt de tip "scurtătură" (Link Table), stabilirea relaţionării se va face numai în baza de date care conţine tabelele originale.

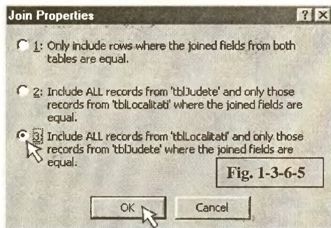


Fig. 1-3-6-5

Butonul "Join Type ..." permite precizarea tipului de relaţie care va fi realizată, astfel:

Stabilirea tipului de relaţie se face prin executarea unui click pe butonul radio situat în partea stângă a explicaţiei, urmat de efectuarea unui click pe butonul OK.

Este admisă şi efectuarea unui dublu click pe butonul radio.

Opțiunea "Only include rows where the ..." stabileşte o relație de tip "unu la unu" între câmpurile celor două tabele, ceea ce înseamnă că unei înregistrări de pe coloana IdJudet a tabelului tblJudete îi va corespunde o singură înregistrare în coloana IdJudet a tabelului tblLocalitati.

Opțiunea "Include ALL records from tblJudete..." stabileşte o relație de tip "Many to One", ceea ce înseamnă că o înregistrare din tabelul tblLocalitati va avea corespondente mai multe înregistrări din tabelul tblJudete.

Pentru situația noastră alege-ți opțiunea a treia "Include ALL records from tblLocalitati..." , ceea ce înseamnă că mai multe înregistrări din tabelul tblLocalitati vor avea cel puțin un corespondent în tabelul tblJudete.

**Pasul 5:** Rezultatul, după apăsarea pe butonul OK, va fi imaginea din figura 1-3-6-6, care vă sugerează cu ajutorul săgeților, tipurile de relații stabilite între cele trei tabele. Din schemă se observă că au fost create două relații de tip "Many to One" – mulți la unul – una între înregistrările coloanei IdJudet din tabelul tblLocalitatati şi înregistrările coloanei IdJudet din tabelul tblJudete, iar alta (pe care o veți crea dumneavoastră) între înregistrările coloanei IdLocalitate a tabelului AdreseElevi şi înregistrările coloanei IdLocalitate a tabelului tblLocalitati.

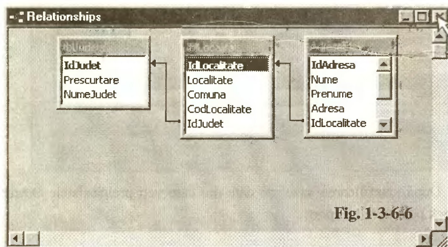


Fig. 1-3-6-6

### 1.3.7. Importul şi exportul datelor

Importul datelor reprezintă o metodă comodă şi rapidă de preluare a datelor din diverse tipuri de fişiere care pot stoca datele şi introducere a acestora în baza de date a utilizatorului.

Dezavantajul importului de date este acela că odată cu preluarea datelor utile se preia şi structura fişierului care conține datele iar utilizatorul va trebui să modifice această structură ulterior pentru a o putea folosi şi adapta cerințelor sale.

**Pasul 1:** Pentru a importa datele stocate într-un alt fişier extern bazei dumneavoastră de date executați un click cu dreapta în fereastra Database iar din meniul de context care se va deschide executați un click pe opțiunea "Import".

Activarea acestei comenzi ar fi putut fi realizată şi prin executarea unui click pe meniul File – Get External Data – Import.

**Pasul 2:** În figura 1-3-7-1 trebuie să precizați numele bazei de date de tip Microsoft Access din care veți prelua datele. Pentru a înțelege funcționarea acestei ferestre de comunicare revedeți explicațiile de la fig. 1-1-3.

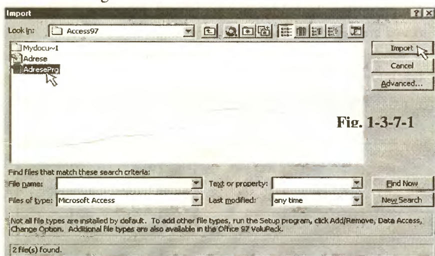
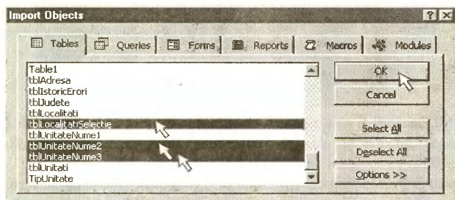


Fig. 1-3-7-1

După identificarea bazei de date din care veți prelua datele executați un click pe butonul Import.

În continuare executați un click pe numele fişierului (tblNume) din baza de date pe care doriți să-l preluați iar apoi un click pe butonul OK pentru începerea transferului datelor.




În urma acestei operații fişierul tblNume va apare în baza dumneavoastră de date, putând fi folosit de acum înainte ca orice alt tabel creat de Microsoft Access pentru bazele sale de date.

**Pasul 1:** Operațiunea inversă importului datelor – exportul – vă permite transferul tuturor lucrărilor realizate (tabele, interogări, formulare, etc.) în baze de date externe sau în fişiere de un alt tip care pot fi folosite şi interpretate de alți utilizatori. Pentru aceasta deschideți meniul de context al oricărui obiect din fereastra Database executând un click cu butonul drept al mouse-ului pe numele acestuia şi activați opțiunea "Save As/Export". Acelaşi rezultat îl puteți obține deschizând meniul File şi alegând opțiunea "Save As/Export" din acesta.

În această situație vom exporta fişierul tblJudete într-un fişier Excel situat în afara bazei noastre de date, deoarece exportul unui tabel sau al unui obiect în interiorul aceleiaşi baze de date înseamnă de fapt copierea lui şi acest lucru nu ne ajută foarte mult în situația de față.

**Pasul 2:** Alegeți prin urmare opțiunea "To an External File or Database ..." din fereastra "Save as..." şi executați apoi un click pe butonul OK, aşa cum se observă în fig. 1-3-7-3.

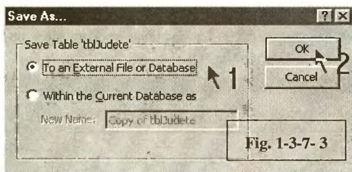


Fig. 1-3-7- 3

Dacă ați fi ales opțiunea "Within the Current Database as" ar fi fost activat imediat și câmpul New Name iar dumneavoastră ați fi avut posibilitatea acordării unui alt nume fișierului nou creat.

**Pasul 3:** Precizați numele fișierului care va fi creat în folderul destinație, (în câmpul File Name), tipul fișierului care va fi creat, alegând din lista

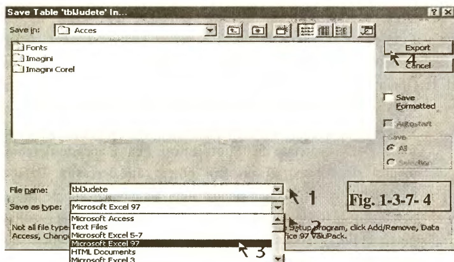


Fig. 1-3-7- 4

derulantă "Save as type..." aplicația corespunzătoare (Excel 97 în cazul nostru) și executați un click pe butonul "Export", așa cum este indicat în fig. 1-3-7-4.

Dacă nu există nici o eroare în funcționarea sistemului dumneavoastră, exportul va fi realizat, iar programul Access 97 va reveni la fereastra Database.

# INTEROGĂRI (QUERY)

## 2.1. Noţiuni generale

Interogările sunt folosite de aplicaţia Access 97 pentru vizualizarea, modificarea şi analizarea datelor în diferite moduri. De asemenea ele pot fi folosite ca o sursă de înregistrări pentru formularele şi rapoartele utilizatorilor.

Aşa cum se observă din figura 2-1-1, interogarea "tblLocalitati_Crosstab" combină informaţiile conţinute în tabelele "tblLocalitati" şi "tblJudete" pentru a oferi un centralizator al localităţilor aparţinând unei comune oarecare dintr-un judeţ înregistrat în baza de date. Din exemplul folosit aici se poate observa că în judeţul Suceava există două localităţi în comuna

tblLocalitati : Table

| IdLocal | Localitate | Comuna | CodLocalitate | IdJudet |
|---|---|---|---|---|
| 55 | Brăiesti | Comu Luncii | 5777 | Suceava |
| 386 | Sasca Mică | Comu Luncii | 5777 | Suceava |
| 385 | Sasca Mare | Comu Luncii | 5777 | Suceava |
| 307 | Păiseni | Comu Luncii | 5777 | Suceava |
| 113 | Măruntisu | C( | | |
| 140 | Dănila | D: | | |
| 396 | Mărginenii de | D: | | |
| 411 | Vlădeni | D: | | |
| 79 | Călinesti Enac | D: | | |

tblJudete : Table

| IdJude | Prescurtare | NumeJudet |
|---|---|---|
| 35 | SM | Satu Mare |
| 36 | SV | Suceava |
| 37 | TL | Tulcea |

tblLocalitati_Crosstab : Crosstab Query

| Comuna | | DB | DJ | GJ | GL | HD | MM | SV | VS |
|---|---|---|---|---|---|---|---|---|---|
| CIPRIAN PORUMBESCU | | | | | | | | 2 | |
| Comu Luncii | | | | | | | | 4 | |
| Comuna | Localitate | 1 | | | | | | | |
| Comu Luncii | Brăiesti | 2 | | | | | | 1 | |
| Comu Luncii | Păiseni | | | | | | | 1 | |
| Comu Luncii | Sasca Mare | | | | | | | 1 | |
| Comu Luncii | Sasca Mică | | | | | | | 2 | |
| Costestii din Vale | Măruntisu | 3 | | | | | | | |
| Dărmănesti | Dănila | | | | | | | 1 | |
| Dărmănesti | Mărginenii d | | | | | | | 1 | |

Fig. 2-1- 1

Ciprian Porumbescu şi patru localităţi în comuna Cornu Luncii – Brăieşti, Păiseni, Sasca Mare şi Sasca Mică.

## 2.2. Tipuri de interogări

În funcţie de tipul informaţiilor oferite, de modul de prelucrare al informaţiei ţi de acţiunea executată, interogările sunt de mai multe tipuri:

1. Interogări de selecţie – "Select Query" – care permit selectarea datelor după un criteriu definit de utilizator, din unul sau mai multe tabele sau chiar din fişiere externe bazei de date. Interogările de selecţie sunt cel mai des folosite.

2. Interogări încrucişate – "Crosstab Query" – care permit selectarea şi prelucrarea datelor după un criteriu definit de utilizator, din unul sau mai multe tabele sau chiar din fişiere externe bazei de date.

3. Interogări pentru crearea tabelelor – "Make Crosstab Query" – care au ca efect crearea unui tabel în care vor fi depuse înregistrările selectate şi prelucrate după un criteriu definit de utilizator.

4. Interogări de adăugare – "Append Query" – care permit adăugarea de noi înregistrări într-un tabel oarecare al unei baze de date.

5. Interogări de actualizare – "Update Query" – care permit actualizarea înregistrărilor unui tabel oarecare al bazei de date, după un criteriu de asemenea definit de utilizator.

6. Interogări de ştergere – "Delete Query" – care permit ştergerea înregistrărilor unui tabel al bazei de date

În Access 97 interogările pot fi folosite ca sursă a unui "set de înregistrări " (Recordset) pentru controalele unui formular sau raport.

## 2.3. Crearea unei interogări

Pentru crearea unei interogări executaţi un click pe eticheta "Queries" a ferestrei bazei de date, astfel ca aceasta să vină în prim plan, iar apoi

executaţi un click pe butonul "New" din fereastra "Database". (Ca variantă de executare a aceleiaşi operaţiuni executaţi un click pe meniul "Insert" al aplicaţiei Access 97 iar apoi un click pe opţiunea Query).

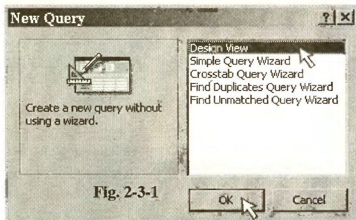


Fig. 2-3-1

În caseta "New Query" care va apare sunteţi invitat să precizaţi (prin executarea unui click pe denumirea din lista oferită) modalitatea de crearea a noii interogări.

Pentru realizarea obiectivelor propuse în acest manual, executaţi un click pe opţiunea "Design View" din listă iar apoi un click pe butonul "OK".

### 2.3.1. Modul Proiectare (Design View)

Dacă aţi executat corect paşii descrişi anterior, pe ecranul monitorului dumneavoastră ar trebui să apară imaginea din figura 2-3-1-1. Sunteţi invitat în acest moment să precizaţi care tabele vor contribui la furnizarea datelor pentru realizarea interogării dumneavoastră.

Furnizarea datelor necesare realizării interogării poate fi realizată fie de tabelele fie de interogările bazei de date.

**Pasul 1:** Pentru aceasta selectaţi, prin executarea unui click pe nume, tabelul sau interogarea sursă şi apăsaţi pe butonul "Add" din caseta de dialog "Show Table". Folosiţi pentru interogarea noastră tabelele "tblJudete" şi "tblLocalităti".

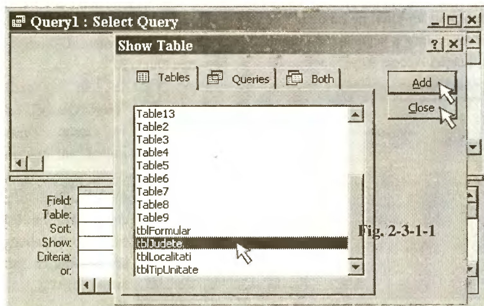


Fig. 2-3-1-1

În situația în care ați dori ca datele necesare interogării dumneavoastră să fie furnizate de o interogare existentă trebuie să efectuați un click pe eticheta “Queries” din caseta de dialog “Show Table” şi să reluați paşii descrişi în paragraful anterior.

**Pasul 2:** Apăsați în continuare pe butonul “Close” pentru a încheia procesul de stabilire a furnizorilor datelor necesare interogării.

Precizarea coloanelor, celor două tabele selectate anterior, care conțin datele care vor fi afişate de către interogarea noastră se face, fie:

**1.** Prin executarea unui dublu click pe numele coloanei unuia din tabele selectate (în exemplul nostru coloana “NumeJudet” din tabelul “tblJudete”)

**2.** Prin “deschiderea” (efectuarea unui click pe secțiunea activă – triunghiul cu vârful orientat în jos) listei derulante situată în secțiunea inferioară a ferestrei “Query1: Select Query” şi efectuarea unui click pe numele coloanei (în cazul nostru pe elementul “tblJudete.NumeJudet”. În această listă derulantă punctul situat între numele tabelului – “tblJudete” – şi numele coloanei acestui tabel este folosit ca separator. Lipsa acestui punct

va face ca aplicaţia Access 97 să interpreteze expresia din câmpul "Field" ca fiind o variabilă definită de utilizator, ceea ce ar putea duce la afişarea unor rezultate greşite de către interogare).

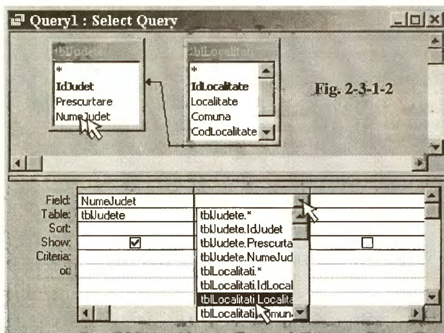


Fig. 2-3-1-2

**3.** Prin agăţarea unei coloane – din caseta corespunzătoare unui tabel cu date sursă afişată în secţiunea superioară a ferestrei "Query1:Select Query" – şi aruncarea acesteia într-unul din câmpurile libere de pe rândul "Field".

Pentru crearea acestei interogări al cărei scop este acela de afişa lista localităţilor înregistrate în baza de date, ordonată în ordinea alfabetică a numelui localităţilor şi grupată după numele judeţelor (şi ele aranjate în ordinea alfabetică a numelui) este necesar, ca folosind una din cele trei variante descrise mai sus – şi sugerate de fig. 2-3-1-2 – să alegeţi pentru primul câmp de pe rândul "Field" coloana "NumeJudet" din tabelul "tblJudet", pentru cel de-al doilea câmp coloana Localitate din tabelul

"tblLocalitati", iar pentru cel de-al treilea câmp coloana "Comuna" din tabelul "tblLocalitati".

Executarea întocmai a operațiunilor descrise în acest capitol ar trebui să conducă la apariția unei ferestre asemănătoare cu cea din fig. 2-3-1-3.

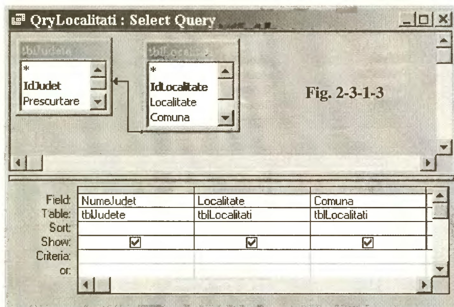


Fig. 2-3-1-3

Stabilirea tipului de sortare a datelor care vor fi afişate pe coloanele interogării se realizează prin alegerea unuia din elementele "Ascending" – pentru sortare ascendentă – sau "Descending" – pentru sortare descendentă. Alegerea opțiunii "not sorted" din această lista derulantă va ca efect afişarea datelor din coloana aleasă în ordinea în care acestea au fost introduse în baza de date. Stabiliți în continuare o sortare de tip ascendent pentru toate cele trei coloane ale interogării.

Stabilirea criteriului după care se va afişarea numai a anumitor date, din totalul celor înregistrate se face prin introducerea unei expresii numerice sau literare în câmpul "Criteria". Introduceți pentru exemplul nostru în câmpul "Criteria" de pe coloana "NumeJudet" expresia "Suceava" (inclusiv ghilimelele), în câmpul "Criteria" de pe coloana "Localitate" expresia > "Br" (inclusiv ghilimelele) iar în câmpul "Criteria" de pe coloana "Comuna"

expresia Is Not Null. Dacă aţi efectuat corect indicaţiile din acest paragraf atunci efectul acestor operaţiuni va fi următorul: vor fi afişate doar satele din judeţul Suceava.

**Pasul 3:** Pentru a "vedea" rezultatul acţiunii unei interogări puteţi întreprinde în continuare una din următoarele acţiuni:

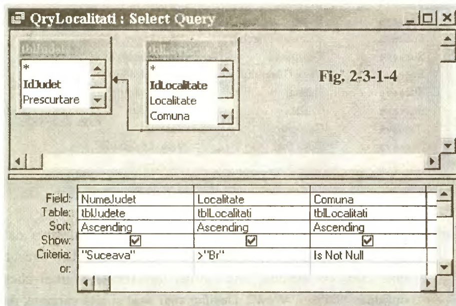


Fig. 2-3-1-4

**1.** Executaţi un click cu dreapta în zona liberă a secţiunii superioare a ferestrei "Query1: Select Query" şi apoi un click pe opţiunea "DataSheet View" din meniul de context.

**2.** Executaţi un click pe meniul "File" al aplicaţiei Access 97 iar apoi un click pe meniul DataSheet View"

**3.** Executaţi un click pe triunghiul cu vârful în jos din partea dreaptă a butonului "View" apoi un click pe opţiunea "DataSheeet View".

Ca urmare a uneia din cele trei acţiuni anterioare alese şi executate de dumneavoastră interogarea creată – căreia i-a fost atribuită deocamdată denumirea generică "Query1" – va fi executată imediat iar rezultatele sale vor fi afişate într-un tabel.(v. fig. 2-1-3-1-5).

QryLocalitati : Select Query

| NumeJudet | Localitate | Comuna |
|---|---|---|
| Suceava | Brăiesti | Cornu Luncii |
| Suceava | Brasca | Ciprian Porumbescu |
| Suceava | Brodina de Sus | Izvoarele Sucevei |
| Suceava | Brosteni | Drăguseni |
| Suceava | Buda | Zvoriste |
| Suceava | Călinesti Enache | Dărmănesti, Măritei |
| Suceava | Călugărita | Horodnic de Sus |
| Suceava | Capu Câmpului | Valea Moldovei |
| Suceava | Colacu | Fundu Moldovei |
| Suceava | Corlata | Drăgoiesti |
| Suceava | Corocăiesti | Veresti |
| Suceava | Cotu Băii | Vadu Moldovei |
| Suceava | Dănila | Dărmănesti |

Record: 1 of 69

Fig. 2-3-1-5

**Pasul 4:** Pentru a reveni la modul proiectare, după vizualizarea datelor sortate şi ordonate după criteriul stabilit recomandăm executarea unui click pe meniul "File" urmat de executarea unui click pe opțiunea "Design View". Desigur mai sunt şi alte variante, dar ce satisfacție ați mai avea dumneavoastră cititorii dacă nu v-ar mai rămâne nimic de descoperit şi explorat la o aplicație atât de complexă ?

Interesant ar fi însă, dacă ați reuşit să ajungeți până la acest punct să analizăm şi o altă opțiune care vă este pusă la dispoziție în meniul "File" al aplicației, şi anume opțiunea "SQL View" – "Structured Query Language"

Dacă veți activa această opțiune, în aceeaşi fereastră în care au fost afişate rezultatele interogării vor fi afişate instrucțiunile VBA (limbajul de programare Visual Basic Access) pe care aplicația le-a folosit la crearea interogării dumneavoastră. Cunoaşterea perfectă a semnificației instrucțiunilor VBA v-ar permite eliminarea tuturor paşilor făcuți în acest capitol pentru realizarea unei interogări, de altfel destul de simplă.

În tabelul următor ne propunem să explicăm instrucțiunile și expresiile folosite de limbajul VBA pentru crearea interogării solicitate.

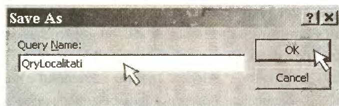
**Observație**: Afișarea instrucțiunilor VBA de către aplicația Access 97 este realizată cu majuscule, iar variabilele interne sunt afișate cu ajutorul expresiilor scrise după regula: prima literă cu literă mare iar următoarele din cadrul acelui grup, cu litere mici.

| | |
|---|---|
| **SELECT** tblJudete.NumeJudet, tblLocalitati.Localitate, tblLocalitati.Comuna **FROM** tblJudete **RIGHT JOIN** tblLocalitati **ON** tblJudete.IdJudet = tblLocalitati.IdJudet **WHERE** (((tblJudete.NumeJudet)="Suceava") **AND** ((tblLocalitati.Localitate)>"Br") **AND** ((tblLocalitati.Comuna) Is Not Null)) **ORDER BY** tblJudete.NumeJudet, tblLocalitati.Localitate, tblLocalitati.Comuna; | |
| **SELECT** | Instrucțiune VBA folosită pentru a preciza că urmează a fi realizată o selecție. |
| tblJudete.NumeJudet, tblLocalitati.Localitate, tblLocalitati.Comuna | Numele coloanelor tabelelor care conțin datele necesare pentru realizarea interogării. Numele tabelelor trebuiesc în mod obligatoriu separate cu ajutorul semnului punct. |
| **FROM** tblJudete | Precizează numele tabelului (sau al interogării!) sau tabelelor (sau interogărilor!) sursă din care vor fi preluate înregistrările necesare interogării. În cazul nostru tabelul "tblJudete" este tabelul master care constituie sursa preluării datelor necesare interogării |
| **RIGHT JOIN** tblLocalitati | Se face precizarea că tabelul master este relaționat la dreapta printr-o relație de tip One to Many cu tabelul "tblLocalitati". |
| **ON** tblJudete.IdJudet = tblLocalitati.IdJudet | Realizarea relației de tip "One to Many" este realizată prin intermediul câmpurilor "IdJudet" prezente în tabelele "tblJudet" și "tblLocalitati". Pe primul loc în cadrul egalității este trecut întotdeauna câmpul indexat al tabelului master. |
| **WHERE** | Se precizează faptul afișarea datelor trebuie să îndeplinească anumite condiții |
| (tblJudete.NumeJudet)="Suceava") | Din tabelul "tblJudete" vor fi preluate numai înregistrările care conțin pe coloana "NumeJudet" expresia "Suceava". |

| | |
|---|---|
| **AND** | Operator logic ŞI. Prezenţa sa în cadrul unei astfel de selecţii impune îndeplinirea, de către setul de înregistrări, atât a condiţiei impuse în partea stângă cât şi a condiţiei impuse în partea dreaptă. |
| (tblLocalitati.Localitate)>"Br") | Se condiţionează în acest fel afişarea localităţilor al căror nume începe cel puţin cu combinaţia de caractere Br. Folosirea operatorilor matematici împreună cu expresii de tip literar este posibilă datorită faptului că VBA efectuează o evaluare valorică a expresiilor literare. |
| **AND** ((tblLocalitati.Comuna) **Is Not Null**)) | Pe lângă cele două condiţii de mai sus mai trebuie ca în câmpurile coloanei "Comuna" din tabelul "tblLocalitati" să se afle cel puţin un caracter. (să nu fie nule). Atenţie: lipsa unei informaţii dintr-un câmp oarecare presupune că acel câmp are o valoare nulă. Astfel introducerea cifrei zero într-un câmp numeric presupune că acel câmp are valoarea zero, care din punctul de vedere al VBA este diferită şi chiar mai mare decât valoarea Null. |
| **ORDER BY** | În traducere "Ordonează după...". |
| tblJudete.NumeJudet, tblLocalitati.Localitate, tblLocalitati.Comuna; | Sunt enumerate coloanele tabelelor sursă care conţin datele ce trebuiesc ordonate după un anumit criteriu. |
| **ASC** | Prezenţa acestei instrucţiuni după coloanele criteriului de ordonare, precizează faptul că ordonarea va fi de tip ascendent. În cazul în care aceasta lipseşte VBA o consideră ca fiind implicită şi o execută. |

Închiderea ferestrei interogării create în modul proiectare echivalează cu terminarea creării interogării. Închiderea se face în mod asemănător modului în care se închide orice fereastră Windows.

După ce aţi apăsat pe butonul de închidere a ferestrei interogării veţi fi întrebaţi imediat dacă doriţi să salvaţi interogarea creată pentru a putea fi folosită ulterior Dacă sunteţi de acord apăsaţi pe butonul Yes şi atribuiţi numele QryLocalitati interogării create.




## 2.3.2. Modul Simple Query Wizard

Reprezintă metoda cea mai simplă pentru crearea unei interogări. Pentru modul intuitiv în care poate fi creată recomandăm folosirea acestei metode, însă pentru posibilitățile reduse pe care le oferă recomandăm folosirea modului de lucru "Design View".

Pentru crearea unei interogări prin această metodă, executați un click pe butonul "New" al ferestrei "Database" atunci când eticheta "Query" este în prim plan. În continuare după ce executați un click pe opțiunea "Simple Query Wizard", nu vă mai rămâne altceva decât să executați un click pe butonul OK.

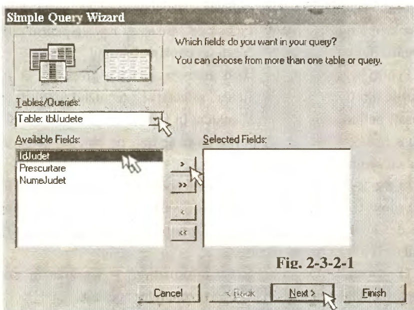



Fig. 2-3-2-1

În fig. 2-3-2-1 veți putea să alegeți, deschizând lista derulantă "Tables/Queries", oricare dintre tabelele sau interogările existente în baza dumneavoastră de date pentru a fi folosită că sursă pentru noua dumneavoastră interogare.. Alegeți pentru exemplul nostru tabelul "tblLocalitati". Aşa cum veți observa imediat după alegerea sursei datelor asistentul Wizard afişează în lista "Available Fields" numele coloanelor care pot furniza datele necesare interogării dumneavoastră. Pentru a stabili câmpurile furnizoare de date puteți să:

**Pasul 1:** Executați un dublu click pe numele câmpului din lista "Available Fields"

**Pasul 2:** Să executați un singur click pe numele câmpului urmat de un click pe butonul "Add" notat în figură cu semnul ">".

Ne propunem să realiză cu ajutorul acestei metode o interogare care să aibă un rezultat asemănător celei din cap. 2-3-1. Pentru aceasta selectați şi transferați în caseta "Selected Fields", câmpurile "Localitate" şi "Comuna".

Revenind la lista derulantă "Tables/Queries" deschideți-o şi alegeți din interiorul acesteia tabelul "tblJudete". După apariția câmpurilor tabelului "tblLocalități" în lista "Available Fields". Transferați în partea dreaptă câmpurile "IdJudet" şi "NumeJudet". Presupunând în acest moment că acestea sunt singurele câmpuri sursă de care aveți nevoie pentru construirea interogării dumneavoastră executați un click pe butonul "Next".

Pentru a renunța, în cazul transferului din greşeală, la unul din câmpurile din lista "Selected Fields" selectați-l executând un click pe acesta şi apăsați pe butonul "Remove" marcat în figura 2-3-2-1 cu semnul "<", iar pentru a renunța la toate câmpurile alese apăsați pe butonul "Remove All", marcat cu semnul "<<".

În fig. 2-3-2-2, care va apare după apăsarea pe butonul "Next" sunteți invitat să precizați numele sub care va fi salvată în baza de date interogarea creată de dumneavoastră (pasul 1). În continuare aveți la dispoziție două opțiuni din care va trebui să alegeți:

**1:** "Open te Query to View information" – lansarea în execuție a interogării create pentru vizualizarea rezultatelor acesteia.

**2:** "Modify the query design" – intrarea în modul de lucru "Design View" pentru a putea observa modalitatea folosită de asistentul Wizard pentru crearea interogării şi pentru eventualele corecturi din partea dumneavoastră.

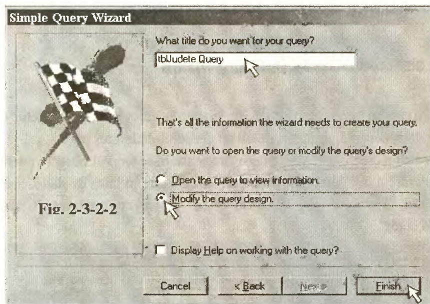


Fig. 2-3-2-2

Bifaţi opţiunea a doua şi apăsaţi pe butonul "Finish".

Interogarea va în acest moment creată şi salvată în baza dumneavoastră de date cu numele "tblLocalitatiQuery", aşa cum probabil l-aţi acceptat şi dumneavoastră iar pe ecran va apare interogarea afişată în modul de lucru "Design". Observaţi că datele nu sunt ordonate după nici un criteriu, dar aceasta nu înseamnă că vi se interzice stabilirea criteriilor de ordonare.

Editaţi interogarea creată şi intraţi în modul "SQL View". Instrucţiunile folosite de asistentul Wizard pentru crearea acestei interogări sunt mult mai puţine.

**SELECT DISTINCTROW** tblLocalitati.Localitate, tblLocalitati.Comuna, tblJudete.IdJudet, tblJudete.NumeJudet

**FROM** tblJudete **RIGHT JOIN** tblLocalitati **ON** tblJudete.IdJudet = tblLocalitati.IdJudet;

Efectul instrucțiunii "**DISTINCT**" care apare aici, spre deosebire de codul sursă al interogării anterioare, este acela de a elimina din lista rezultatelor afişate înregistrările cu acelaşi conținut.

În final închideți interogarea apăsând pe butonul "Close" situat în colțul dreapta sus al ferestrei în care a fost deschisă.

### 2.3.3. Modul Crosstab Query Wizard

Permite combinarea rezultatelor stocate în înregistrările tabelelor unei baze de date, permițând uneori rezultate deosebit de interesante.

Pentru a crea o astfel de interogare (care în exemplul nostru va realiza o centralizare a din comunele județelor înregistrate în baza de date), apăsați pe butonul "New" al ferestrei "Database" şi alegeți opțiunea "Crosstab Query Wizard".

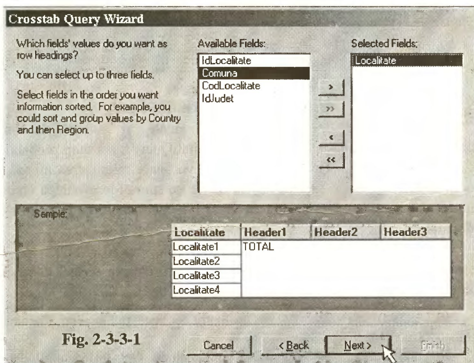



Fig. 2-3-3-1

**Pasul 1:** În primul pas asistentul Wizard vă cere să precizaţi numele tabelului care conţine înregistrările ce vor fi afişate pe rândurile interogării ce va fi creată. Alegeţi prin executarea unui click pe numele afişate în listă tabelul “tblLocalitati” ca sursă pentru rândurile interogării şi apăsaţi pe butonul “Next” pentru a trece la pasul următor.

**Pasul 2:** În pasul următor va trebui să precizaţi numele coloanelor din tabelul sursă care conţin valorile ce se vor constitui în titlul rândurilor interogării noastre (fig. 2-3-3-1).

Alegeţi prin executarea unui dublu click, câmpul “Localitate” din lista câmpurilor. În această situaţie orice greşeală nu mai poate fi corectată decât prin întoarcerea la pasul anterior – lucru pe care îl puteţi realiza executând un click pe butonul “Back”.

**Pasul 3:** În pasul următor asistentul Wizard vă cere să precizaţi numele coloanei din tabelul sursă ale cărui valori se vor constitui în anteturile coloanelor interogării. Stabiliţi acest lucru prin executarea click (pe câmpul “IdJudet”) urmat de încă un click pe butonul “Next” (fig. 2-3-3-2).

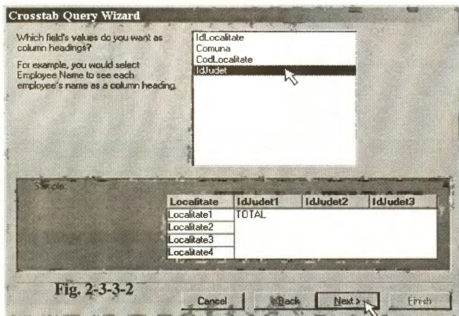



Fig. 2-3-3-2

**Pasul 4:** În acest ecran trebuie să stabiliți expresia care va fi calculată și afișată la intersecția unei coloane și a unui rând al interogării dumneavoastră. Pentru aceasta asistentul Wizard vă pune la dispoziție un set de funcții standard – pe care le veți putea modifica ulterior conform necesităților dumneavoastră.

**Avg** Calculează media aritmetică a valorilor înscrise în câmpul IdLocalitate.

**Count** Returnează numărul înregistrărilor ne-nule din câmpul ales.

**Max** Returnează cea mai mare valoare din câmpul ales.

**Min** Returnează cea mai mică valoare din câmpul ales.

**Sum** Returnează suma valorilor din câmpul ales.

**Var** Returnează variația populației dintr-un set de înregistrări.

**StDev** Returnează deviația standard a unei populații dintr-un set de înregistrări.

**First** Returnează prima valoare a unui set de înregistrări.

**Fig. 2-3-3-3**

Alegeţi pentru exemplul nostru funcţia “Count” – contorizare valori nenule – iar dacă doriţi însumarea valorilor de pe fiecare rând bifaţi caseta de validare “Yes, include row sums” şi apăsaţi pe butonul “Next” pentru continuare.

**Pasul 5:** În pasul următor trebuie să precizaţi numele interogării create (acordaţii numele “tblLocalitati_Crosstab1”) şi evenimentele care se vor desfăşura după crearea interogării – alegeţi opţiunea “View the query” şi apăsaţi pe butonul “Finish”.

tblLocalitati_Crosstab2 : Crosstab Query

| Localitate | Total Of IdLocal | 3 | 5 | 7 |
|---|---|---|---|---|
| Voinesti | 1 | | | |
| Voivodeasa | 1 | | | |
| Voivodeni | 1 | | | |
| Volovăt | 1 | | | |
| Vornicenii Ma | 1 | | | |
| Vrâncioaia | 1 | | | |
| Vulcan | 2 | | | |
| Vulcana Pand | 1 | | | |
| Vulturu | 1 | | | |
| Zalău | 1 | | | |
| Zam | 1 | | | |
| Zamostea | 1 | | | |
| Zărneşti | 1 | | | |
| Ziduri | 1 | | | |
| Zizin | 1 | | | |

Record: 529 of 538

Fig. 2-1-3-3-5

Aşa cum se observă din tabelul de mai sus numele judeţelor este reprezentat de codurile numerice unice ale acestora şi afişat în antetul coloanelor. La nivel de rânduri, interogarea a realizat acolo unde a fost cazul, o centralizare a numelui localităţilor, fără a pierde însă apartenenţa la un anumit judeţ a acestora. Astfel localitatea “Vulcan” apare de două ori în baza de date, o dată în judeţul cu numărul unic de identificare 10, şi încă o dată, deşi nu este vizibilă, în judeţul cu numărul unic de identificare 22. Dacă veţi derula în sus informaţiile din tabel, veţi observa că interogarea

creată a mai realizat o centralizare la nivelul judeţelor, pe primul rând al tabelului, a localităţilor oferindu-ne în acelaşi timp şi un total al numelor distincte de localităţi – ceea ce în acest caz poate fi considerat o greşeală destul de gravă, pentru că este admis faptul că pot exista două sau mai multe localităţi cu acelaşi nume situate însă în judeţe diferite.

Dacă aţi reuşit să ajungeţi la acest ecran, reveniţi în modul proiectare ("Design View"), pentru a face corecturile necesare şi pentru a îmbunătăţi performanţele interogării.

Modificaţi, conform figurii următoare (fig. 2-3-3-5), structura interogării noastre.

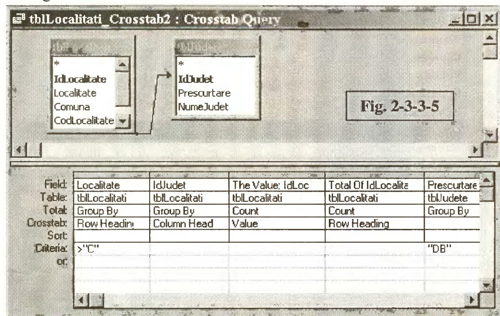


Fig. 2-3-3-5

După ce aţi modificat interogarea conform indicaţiilor din figura de mai sus reveniţi în modul "DataSheet View" pentru a vizualiza rezultatele centralizate astfel, iar această dată veţi aprecia acurateţea informaţiilor.

Dacă însă în câmpul "Criteria", sub coloana "Prescurtare", aţi introduce expresia "DB" –de exemplu – iar în câmpul "Criteria", sub coloana "Comuna", expresia > "C", interogarea ar afişa doar lista

localităţilor al căror nume începe cel puţin cu litera "C" şi care aparţin judeţului Dîmboviţa.

În final salvaţi această interogare sub numele de "tblLocalitati_Crosstab".

### 2.3.4 Modul Find Duplicates Query Wizard

Vă permite aflarea şi afişarea înregistrărilor cu acelaşi conţinut. Pentru aceasta porniţi la crearea acesteia la fel ca în exemplul anterior apăsând pe butonul "New" din fereastra "Database" şi alegând de această dată opţiunea "Find Duplicates Query Wizard". Executaţi un click pe butonul OK pentru a trece la pasul următor.

În acest moment vi se va cere de către asistentul Wizard să precizaţi numele tabelului sau interogării despre care presupuneţi că ar conţine înregistrări cu un conţinut asemănător.

**Pasul 1:** Executaţi un click pe tabelul "tblLocalitati", urmat de un click pe butonul "Next".

**Pasul 2:** Transferaţi în lista din partea dreaptă câmpurile: "Localitate" şi apăsaţi pe butonul "Next" pentru a trece la ultima fază a creării interogării.

**Pasul 3:** De regulă pe coloana unui tabel pot exista mai multe înregistrări cu acelaşi conţinut- ceva însă le face să deţină în continuare un statut de înregistrare unică în cadrul aceluiaşi tabel, şi anume conţinutul celorlalte informaţii de pe acelaşi rând. Pentru a verifica acest lucru transferaţi în caseta din partea dreaptă ("Additional _Query Fields") câmpurile "Comuna" şi "IdJudet" şi apăsaţi pe butonul "Next".

În acest fel aţi ajuns la ultimul pas al creării interogării în care sunteţi invitat să introduceţi numele sub care va fi salvat interogarea creată. Lăsaţi numele pe care vi-l propune asistentul Wizard "Find duplicates for

tblLocalitati", bifați, dacă nu este deja bifată opțiunea "View Data" și apăsați pe butonul "Finish".

Rezultatul obținut este suficient de sugestiv reprezentat (v. fig. 2-3-4-1) astfel încât să vă dați seama clar dacă ați introdus de mai multe ori aceeași înregistrare sau dacă într-adevăr înregistrările au fost corect operate.

Find duplicates for tblLocalit

| Localitate | Comuna | IdJudet |
|---|---|---|
| Bârsesti | | Vrancea |
| Boroaia | | Suceava |
| Boroaia | | Suceava |
| Breaza | | Buzău |
| Breaza | | Suceava |
| Buda | | Buzău |
| Buda | Zvoriste | Suceava |
| Dărmănesti | Măritei | Suceava |
| Dărmănesti | | Dâmbovița |
| Paltin | Vatra Moldovit | Suceava |
| Paltin | | Vrancea |
| Putna | | Suceava |
| Putna | | Suceava |

Record: 14 of 27

Fig. 2-3-4-1

Așa cum se observă din fig. 2-3-4-1 prezența a două localități cu numele "Paltin" este normală, deoarece este ceva diferit în informația înregistrată și prezentată – și anume județul – dar prezența a două localități "Putna" în același județ și cu același cod poștal este mai mult ca sigur o greșeală de introducere a datelor.

Așa cum ați observat de fiecare dată când ați început să creați o interogare în lista care vă apare de fiecare atunci când intervine asistentul Wizard pentru crearea unei interogări mai aveți la dispoziție și opțiunea "Find Unmatched Query" Wizard. Această opțiune pe care o veți studia fără ajutorul nostru vă va permite aflarea înregistrărilor "orfane" dintr-un tabel al unei baze de date. Înregistrările orfane în cazul nostru sunt înregistrările care nu au un corespondent în tabelul relaționat sau care conțin valori nule.

## 2.4. Conversia interogărilor

Pentru a observa în ce mod interogările create pot fi modificate astfel încât să servească într-o măsură cât mai mare necesitățile utilizatorilor deschideți interogarea creată în cap. 2.3.1 ("QryLocalitati") în modul de

lucru "proiectare" executând un click pe numele acesteia urmat de un click pe butonul "Design" de pe suprafaţa ferestrei "Database".

Odată intraţi în modul de lucru pentru a transforma interogarea deschisă într-una de un alt tip, executaţi un click pe meniul "Query" al aplicaţiei şi observaţi variantele de interogare în care poate fi transformată.

### 2.4.1. Trecerea la o interogare încrucişată

Dacă aţi executat paşii din cap. anterior (cap. 2.4.) executaţi un click pe opţiunea "Crosstab Query" din meniul "Query".

În urma acestei operaţii între rândurile "Total" şi "Sort" va fi inserat în mod automat rândul "Crosstab". Pentru ca interogarea să se comporte ca o interogare încrucişată nu uitaţi să alegeţi pe rândul "Crosstab", expresiile "Column Heading", "Row Heading" şi "Value" cel puţin o singură dată şi să adăugaţi la interogare o nouă coloană având ca sursă aceleaşi înregistrări pe care le-aţi folosit pe coloana pe care aţi ales expresia "Value" – valoare. Este indicat de asemenea ca pe coloana ce conţine valorile necesare interogării să nu existe nici un criteriu de sortare sau de afişare condiţionată a înregistrărilor. În final salvaţi interogarea folosind însă un alt nume.

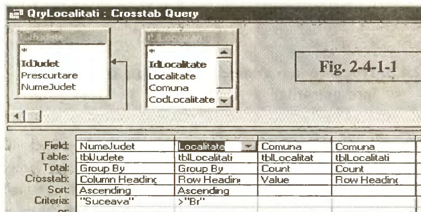


Fig. 2-4-1-1

## 2.4.2. Trecerea la o interogare care permite crearea unui tabel în baza de date

După executarea paşilor prevăzuţi la cap. 2.4. alegeţi opţiunea “Make-Table Query”. şi în continuare precizaţi numele tabelului (“tblTest”) în care vor fi adăugate rezultatele interogării dumneavoastră.

În urma acestei operaţii, la lansarea în execuţie a acestei interogări, în baza dumneavoastră de date va fi creat un tabel cu numele “tblTest” care va conţine toate înregistrările selectate de interogare.

Pentru această interogare codul VBA se prezintă astfel:

```
SELECT tblJudete.NumeJudet, tblLocalitati.Localitate,
tblLocalitati.Comuna INTO tblTest FROM tblJudete RIGHT JOIN tblLocalitati
ON tblJudete.IdJudet = tblLocalitati.IdJudet WHERE
(((tblJudete.NumeJudet)="Suceava") AND (tblLocalitati.Localitate) >"Br") AND
((tblLocalitati.Comuna) Is Not Null)) ORDER BY tblJudete.NumeJudet,
tblLocalitati.Localitate, tblLocalitati.Comuna;
```

Observaţi că, spre deosebire de codul sursă al interogării “QryLocalitati” nu apare în plus decât instrucţiunea **“INTO”** urmată de numele tabelului care va fi creat şi în care vor fi preluate rezultatele interogării.

## 2.4.3. Trecerea la o interogare de actualizare a înregistrărilor

Interogările de actualizare sunt extrem de utile atunci când se intenţionează modificarea după un anumit criteriu a datelor înregistrate. Dacă de exemplu în rezultatele preluate dintr-un tabel şi afişate, numele unei localităţi apare scris cu litere mici, iar dumneavoastră doriţi ca numele localităţilor înregistrate în baza de date să fie memorat cu majuscule, atunci

pe rândul “Update To:”, de pe coloana “Localitati” va trebui să introduceți expresia: Format(‘Localitate’,’>’).

**Rețineți**: actualizarea datelor nu se va produce în momentul vizualizării rezultatelor ci în momentul lansării în execuție a interogării, iar operațiunea nu este reversibilă și în această situație orice greșeală vă va costa chiar datele înregistrate.

| | | | |
|---|---|---|---|
| Field: | NumeJudet | Localitate | Comuna |
| Table: | tblJudete | tblLocalitati | tblLocalitati |
| Update To: | | Format('Localitate';'>') | |
| Criteria: | "Suceava" | >"Br" | Is Not Null |
| or: | | | |

**Fig. 2-4-4-1**

În final nu uitați să salvați și această interogare, bineînțeles cu un alt nume, pentru a putea fi studiată mai târziu.

## 2.4.4. Trecerea la o interogare de adăugare a înregistrărilor

Pentru a realiza adăugarea datelor selectate cu ajutorul unei interogări într-un tabel oarecare al bazei de date, în afară de introducerea manuală a datelor, se poate apela la ajutorul unei interogări de adăugare. Deschideți pentru exemplificare, interogarea “QryLocalitati”, și din meniul “Query” alegeți opțiunea “Append Query...”. Veți fi întrebat imediat care este numele tabelului în care doriți să fie introduse datele selectate de interogare – fie acesta tabelul “xxx”, care a fost obținut prin copierea tabelului “tblLocalitati” și lipirea sub numele “xxx”.

| | | | |
|---|---|---|---|
| Field: | NumeJudet | Localitate | Comuna |
| Table: | tblJudete | tblLocalitati | tblLocalitati |
| Sort: | Ascending | Ascending | Ascending |
| Append To: | | Localitate | Comuna |
| Criteria: | xxx.* | xxx.* | xxx.* |
| or: | IdLocalitate | IdLocalitate | IdLocalitate |
| | Localitate | Localitate | Localitate |
| | Comuna | Comuna | Comuna |
| | CodLocalitate | CodLocalitate | CodLocalitate |
| | IdJudet | IdJudet | IdJudet |

Precizaţi în continuare coloanele tabelului “xxx” în care vor fi adăugate rezultatele interogării. Aveţi grijă ca atât coloanele tabelului destinaţie cât şi coloanele interogării să fie formatate pentru acelaşi tip de date, în caz contrar, în momentul lansării în execuţie a acestei interogări va fi generat un mesaj de eroare care va bloca desfăşurarea procesului.

În exemplul nostru coloana “Localitate” a tabelului “xxx” va primi datele de pe coloana cu acelaşi nume al interogării, iar coloana “Comuna” a tabelului “xxx” va primi datele de pe coloana “Comuna” a interogării. Se observă de asemenea că rezultatele din coloana “NumeJudet” a interogării nu vor fi adăugate în nici una din coloanele tabelului “xxx” deoarece la proiectarea bazei de date am considerat inutilă repetarea anumitor date în diferite tabele – numele judeţelor fiind memorat o singură dată pe coloana “NumeJudet” a tabelului “tblJudete”.

Codul sursă al acestei interogări este:

```
INSERT INTO xxx (Localitate, Comuna) SELECT tblLocalitati.Localitate,
tblLocalitati.Comuna FROM tblJudete RIGHT JOIN tblLocalitati ON
tblJudete.IdJudet = tblLocalitati.IdJudet WHERE
(((tblJudete.NumeJudet)="Suceava") AND ((tblLocalitati.Localitate)>"Br") AND
((tblLocalitati.Comuna) Is Not Null)) ORDER BY tblJudete.NumeJudet,
tblLocalitati.Localitate, tblLocalitati.Comuna;
```

Nu uitaţi să salvaţi şi această interogare sub un nume diferit e cel al interogării iniţiale.

### 2.4.5. Trecerea la o interogare de ştergere a înregistrărilor

Repetaţi paşii, alegând însă pentru crearea unei astfel de interogări opţiunea “Delete Query”. Reţineţi că la fel ca şi interogarea de actualizarea operaţiunea de ştergere a datelor cu ajutorul interogărilor este ireversibilă, deci asiguraţi-vă că datele care vor fi şterse reprezintă ceea ce aveţi dumneavoastră în plan.

*Capitolul 3*

# FORMULARE (FORMS)

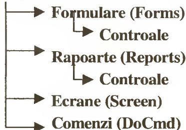
## 3.1. Noţiuni generale

O aplicaţie tipică Windows afişează unul sau mai multe ecrane pline cu obiecte - meniuri, butoane, liste derulante, casete de editare, etc. - cu care utilizatorul va interacţiona pentru a controla evoluţia programului. Obiectele de pe suprafaţa acestor ecrane vor reacţiona într-un mod specific sub acţiunea anumitor evenimente întreprinse de utilizator sau declanşate de o anumită secţiune a programului. Totuşi, răspunsul acestor obiecte la acţiunea utilizatorului este uneori unul standard pentru tipul de obiect asupra căruia s-a acţionat, şi chiar inadecvat cerinţelor utilizatorului.

Pentru a produce un program care să nu fie banal, trebuie să evitaţi sau să extindeţi acest comportament limitat. Puteţi face aşa ceva scriind propriul dumneavoastră cod şi ataşându-l evenimentului vizat al obiectului ţintă. De exemplu evenimentul "Click" al unui buton de comandă ar putea să conţină o singură instrucţiune VBA: "End". Atunci când se va executa un click pe acest buton, va fi declanşat codul asociat acestui eveniment astfel încât în cazul de faţă, programul se va încheia.

O aplicaţie Microsoft Access 97 are următoare structură:

**Aplicaţia**



. În Access 97 formularele, rapoartele şi controalele alcătuiesc bazele vizuale ale unei aplicaţii Windows. Aspectul şi funcţionalitatea acestor controale pot fi controlate şi îmbunătăţite prin modificarea proprietăţilor (care determină caracteristicile unui obiect) şi metodelor (care reprezintă procedurile sau funcţiile care informează un obiect în legătură cu modul de a efectua o anumită acţiune). Modificarea proprietăţilor şi metodelor unui obiect poate fi modificată cu ajutorul liniilor de cod folosindu-se **notaţia cu punct.**

### 3.1.1. Proprietăţi

Stricto senso o proprietate este un atribut nominalizat al unui obiect programabil. Proprietăţile definesc caracteristicile unui obiect – cum ar fi culoarea, dimensiunile, poziţia pe ecran – sau uneori modul în care se comportă un obiect, dacă, de exemplu, o casetă de text acceptă sau nu linii multiple sau dacă un element este activ la un moment dat.

### 3.1.2. Metode

Pentru a defini un obiect nu este suficientă menţionarea proprietăţilor sale, deoarece acestea nu sunt singurele sale caracteristici. O definiţie completă a unui obiect presupune pe lângă aceste caracteristici şi o enumerare a activităţilor pe care obiectul le poate realiza. În programarea orientată spre obiecte o metodă este o procedură asociată sau încorporată, un bloc de instrucţiuni în cod VBA, care poate fi invocat pentru a asocia o anumită acţiune unui anumit obiect. Spre deosebire de alte funcţii şi proceduri ale limbajului VBA, metodele trebuie să fie asociate unui obiect pentru a se afla într-un anumit context.

### 3.1.3. Notația cu punct

Atunci când se lucrează în cod , referirea la proprietățile şi metodele unui obiect se face utilizând notația cu punct. Sintaxa pentru stabilirea proprietății unui obiect este:

Obiect.Proprietate = Valoare

Să presupunem că avem un formular cu numele *MyForm* şi că dorim să stocăm valoarea curentă a proprietății "culoare de fond", apoi să modificăm această culoare. Putem folosi pentru aceasta următoarele linii de cod:

VecheaCuloare = MyForm.BackColor
*Am stabilit în acest moment culoarea de fond.*

MyForm.BackColor = QBColor(Red)
*Am stabilit noua culoare de fond –roşu*

MyForm.BackColor = VecheaCuloare
*Am revenit la vechea culoare.*

### 3.1.4. Convenții de notație

Lucrul în cod presupune lucrul cu foarte multe variabile, proprietăți şi metode. Denumirile acestora pot deveni incomode la un moment dat dacă nu veți stabili un set de reguli pentru numele acestora care să vă permită să recunoaşteți rapid tipul obiectului la care se face referire. Convenția unanim acceptată este aceea de a acorda un prefix din trei litere mici; în felul acesta veți şti imediat la ce tip de obiect se face referire.

| Obiect VBA | Prefix |
|---|---|
| Formular (Form) | Frm |
| Buton de comandă (Command Button) | Cmd, btn |
| Etichetă (Label) | Lbl |
| Casetă de text (TextBox) | Txt |
| Bare de derulare (ScrollBar) | Hsb, vsb |
| Meniu (Menu) | Mnu |
| Butoane de opțiuni (OptionButton) | Opt |
| Casetă de validare (CheckBox) | Chk |
| Casetă de validare (ListBox) | Lst |
| Casetă combinată (ComboBox) | Cbo |
| Cadru (Frame) | Fra |
| Imagine (Image) | Img |
| Figura geometrică (Shape) | Shp |
| Linie (Line) | Lin |
| Orologiu (Timer) | Tmr |

## 3.2. Tipuri de controale

În general controalele disponibile în Access 97 sunt disponibile prin intervenția unor asistenți Wizard care vă ajută să le configurați și să le atribuiți anumite combinații de linii de cod. De multe ori este suficient datorită intervenției acestor asistenți, să recunoașteți doar pictogramele atașate acestora pentru a putea crea un formular care să rezolve destul de bine necesitățile unui utilizator destul de pretențios.

| Control | Nume control | Control | Nume control |
|---|---|---|---|
| | Indicator | | Etichetă |
| | Casetă de text | | Cadru |
| | Grup de butoane 3D | | Buton de opțiune |
| | Casetă de validare | | Casetă combinată |
| | Casetă cu listă | | Buton de comandă |
| | Casetă imagine Picture Box | | Lista Tabbed |
| | Subformular | | Linie |
| | Dreptunghi (Rectangle) | | Bară de derulare |
| | Buton Spinner | | Bară de progres |

Asistenții Wizard nu sunt însă suficient de "inteligenți" pentru a crea controale suficient de sofisticate. În astfel de situații se impune intervenția programatorilor sau a utilizatorilor pentru atașarea de secvențe de cod controalelor unui formular. Instrucțiunile de cod VBA și sintaxa acestora vor fi tratate pe larg în capitolul 5 al acestei lucrări. În continuare ne vom concentra doar asupra folosirii asistenților pentru crearea unor formulare funcționale .

## 3.3. Crearea unui formular

Înainte de a porni la crearea unui formular sau a unui raport trebuie să aveți foarte clar în minte problemele pe care acesta trebuie să le rezolve și modul în care doriți afișarea datelor în cadrul formularului. În funcție aceste lucruri veți putea ști ce tabele sau interogări trebuie să folosiți pentru formularul dumneavoastră. În acest capitol vom crea diverse tipuri de formulare care vor rezolva, sau măcar vor încerca să rezolve aceeași

problemă: afişarea pe ecran a localităţilor şi judeţelor înregistrate în baza de date după un anumit criteriu.

Dacă aveţi deschisă pe ecran aplicaţia Access 97 împreună cu fereastra bazei dumneavoastră de date, pentru a crea un formular nou recomandăm deschiderea meniului "Insert" al aplicaţiei şi alegerea opţiunii "Form".

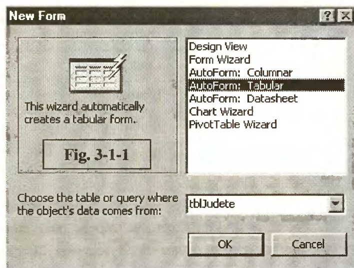


Fig. 3-1-1

Puteţi declara crearea unui nou formular şi prin apăsarea pe butonul "New" de pe suprafaţa ferestrei Database dacă în prim planul ferestrei se află eticheta "Forms".

Răspunsul imediat al asistentului Wizard vă solicita precizarea tabelului sursă a datelor formularului. Deocamdată nu veţi putea preciza decât un singur tabel ca sursă a datelor, însă pe parcurs veţi putea să stabiliţi un criteriu mai complicat pentru crearea sursei datelor acestui formular. Alege-ţi ca sursă principală a datelor tabelul tblJudete aşa cum am procedat şi noi în fig. alăturată. În continuare va trebui să precizaţi tipul formularului pe care doriţi să îl creeze asistentul Wizard pentru dumneavoastră. Alegeţi deocamdată opţiunea "AutoForm: Tabular".

Această fereastră va constitui punctul de plecare pentru construirea tuturor formularelor bazei de date, dar şi pentru construirea tuturor formularelor necesare pentru exemplificarea lucrării, de aceea la crearea celorlalte formulare nu vom menţiona modalitatea de declarare a unui nou formular.

Microsoft Access

File Edit View Insert Format Records Tools Window Help

tblJudete

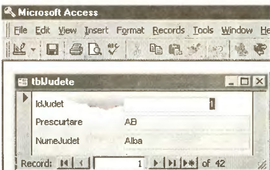
| IdJudet | Prescurtare | NumeJudet |
|---|---|---|
| 1 | AB | Alba |
| 2 | AG | Argeş |
| 3 | AR | Arad |
| 4 | B | Bucureşti |
| 5 | BC | Bacău |
| 6 | BH | Bihor |
| 7 | BN | Bistriţa Năsăud |
| 8 | BR | Brăila |
| 9 | BT | Botoşani |
| 10 | BV | Braşov |
| 11 | BZ | Buzău |

Record: 1 of 42

Fig. 3-1-2

### 3.3.1. Modul Autoform: Columnar

Dacă aţi apăsat deja pe butonul Ok din fig. 3-1-1 atunci, după câteva momente de lucru din partea asistentului pe ecran va apare primul dumneavoastră formular în forma prezentată în fig. 3-1-2. În acest tip de formular înregistrările sunt aranjate pe



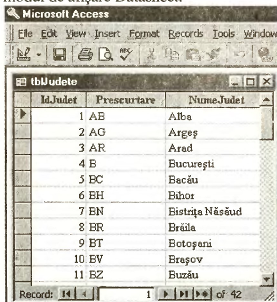
verticală, de-a lungul marginii din stânga a paginii, în ordinea în care apar în definiția tabelului. Toate câmpurile sunt etichetate. La un moment dat, este afişată o singură înregistrare; există şi butoane de navigare, care asigură deplasarea de la o înregistrare la alta.

### 3.3.2. Modul Autoform: Tabular

Înregistrările sunt aranjate într-o "grilă" dispusă pe suprafața paginii vizibile. Câmpurile sunt etichetate. Formularul afişează mai multe înregistrări simultan, iar dacă este necesar include şi butoane de navigare şi bare de derulare orizontale sau verticale.

### 3.3.3. Modul Autoform: Datasheet

Modalitatea de crearea a unui astfel de formular este identică modalității prin care au fost create celelalte două tipuri de formulare Columnar şi Tabelar. Înregistrările sunt reprezentate la fel ca un tabel în modul de afişare Datasheet.




| IdJudet | Prescurtare | NumeJudet |
|---|---|---|
| 1 | AB | Alba |
| 2 | AG | Argeş |
| 3 | AR | Arad |
| 4 | B | Bucureşti |
| 5 | BC | Bacău |
| 6 | BH | Bihor |
| 7 | BN | Bistrița Năsăud |
| 8 | BR | Brăila |
| 9 | BT | Botoşani |
| 10 | BV | Braşov |
| 11 | BZ | Buzău |

Numele de câmpuri se găsesc în partea superioară a fiecărei coloane. Sunt afişate toate înregistrările care încap în formular, linie cu linie; de asemenea sunt create butoane de navigare şi bare de derulare orizontale şi verticale.

### 3.3.4. Modul Form Wizard

Reprezintă un set util de casete de dialog în care sunteți întrebat de unde provin datele, cum doriți să le aranjați în cadrul formularului şi ce imagine grafică doriți să apară în fundal.

Cu ajutorul acestei metode puteți crea foarte rapid un formular cu un aspect plăcut, parcurgând un număr relativ mic de paşi.

Dezavantajul acestei metode este lipsa controlului direct asupra machetei şi asupra legăturilor controalelor grafice din formular. Form Wizard nu creează casete combinate, casete cu liste sau liste de opțiuni şi nu inserează imagini grafice sau chenare. Dacă nu doriți să rămâneți la rezultatele schițate de asistentul "Expert Wizard" va trebui să modificați formularul creat.

Form Wizard
Which fields do you want on your form?
You can choose from more than one table or query.
Tables/Queries:
Table: tblJudete
Available Fields:
Prescurtare
NumeJudet
Selected Fields:
IdJudet
Cancel
< Back
Next >
Finish

Dacă ați ales opțiunea "Form Wizard" pentru crearea formularului, în primul pas va trebui să precizați numele tabelului sursă și câmpurile din acesta care conțin datele pe care le doriți afișate în formular.

Pentru exemplul nostru transferați toate câmpurile din lista "Available Fields" în lista "Selected ields" și apăsați pe butonul "Next".

În pasul următor va trebui să precizați tipul formular pe care asistentul Wizard îl va crea pentru dumneavoastră. Pentru a stabili acest lucru aveți la dispoziție un grup de opțiuni asemănător cu cel din figura următoare:




Va trebui în acest moment să alegeți opțiunea "Justified" deoarece variantele celelalte vor crea formulare identice ca formă cu cele create în capitolele anterioare. Puteți face acest lucru executând un click pe butonul alb din fața fiecărei opțiuni sau puteți apăsa o combinație de taste formată din tasta ALT și una din literele subliniate (tastă scurtătură) din cadrul fiecărei opțiuni. (în situația de față ALT și J). La fel ca în cazul celorlalți asistenți Wizard puteți renunța oricând la proces apăsând pe butonul "Cancel" sau puteți reveni la unul din pașii anteriori apăsând pe butonul "Back". Treceți însă la pentru continuare la pasul următor apăsând pe butonul "Next".




Alegerea unei imagini de fundal pentru formularul creat se face prin alegerea uneia din expresiile din interiorul listei. Dacă doriți ca și în formularul dumneavoastră să apară aceeași imagine de fundal ca și în exemplele folosite de noi alegeți opțiunea "Clouds".

Alegerea uneia din opțiunile corespunzătoare unei anumite imagini de fundal va fi urmată de apariția imaginii alese în caseta de previzualizare din partea stângă.

Odată aleasă imaginea dorită nu vă mai rămâne decât să apăsați pe butonul "Next".

Denumiţi formularul creat frmtblJudeteWizard, bifaţi opţiunea "Open the form view or enter information" şi executaţi un click pe butonul "Finish".

Formularul creat pe această cale va arăta asemeni celui din figura alăturată. Chiar dacă nu vă satisfac pe deplin variantele propuse până acum de asistenţii Wizard, nu disperaţi, veţi putea mai târziu să modificaţi atât aspectul cât şi modul în care se realizează prelucrarea datelor de către formulare.

| IdJudet | Prescurtare | NumeJudet |
|---|---|---|
| 1 | AB | Alba |

Record: 1 of 42

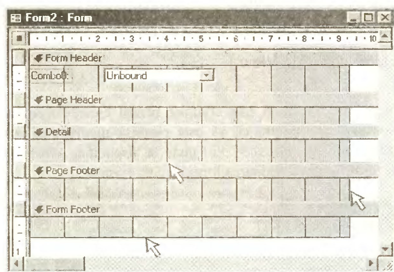
### 3.3.5. Modul Design View

Este modul de lucru cel mai complet pentru crearea formularelor aplicaţiei dumneavoastră pentru că vă este permisă modificarea oricărui element pe care aplicaţia Access 97 vi-l pune la dispoziţie. Acest mod de lucru poate fi folosit şi pentru editarea formularelor existente.

Fig. 3-1-5-1

Pentru a înțelege modul în care funcționează aceste formulare ar trebui să stiti ca datele înregistrate în diferitele tabele ale bazei de date concură pentru realizarea unei interfețe grafice (formulare în cazul nostru). Modalitatea în care datele sunt afişate şi prelucrate în formulare este la îndemâna utilizatorului. Înainte de a trece la realizarea formularului de mai sus va trebui să facem câteva precizări.

### 3.3.5.1. Secțiunile unui formular




**Detail (Detalii).**

Această secțiune a formularului este cea în care apar datele pentru fiecare înregistrare din sursa de date. În formularul creat datele depuse în această secțiune vor apare "suprapuse" – aranjate unele sub altele – putându-se trece însă de la o înregistrare la alta cu ajutorul butoanelor de navigare. Secțiunea Detalii, spre deosebire de celelalte este afişată implicit în toate formularele pe care le creați sau pe care le editați. Ascunderea acestei secțiuni poate fi realizată atunci când interesele utilizatorilor o cer, prin micşorarea dimensiunilor sale până la mărimea unei linii, care nu va fi observată în formular.

**Form Header/Footer (Antetul / Subsolul de formular).**

Aceste secțiuni nu apar în mod implicit atunci când editați un formular, iar dacă aveți nevoie de ele va trebui să solicitați afișarea lor, prin deschiderea meniului "View" al aplicației și bifarea opțiunii Form Header/Footer. În zona de antet de formular se introduc de regulă, informații care nu se modifică de la o înregistrare la alta, cum ar fi: numele unității sau instrucțiuni de utilizare a formularului. Zona de subsol a formularului conține de regulă, informații despre data și ora la care a fost creat formularul, date centralizate, totaluri, informații despre autori.

**Page Header/Footer (Antetul/Subsolul de pagină).**

Aceste secțiuni au rolul de a insera întreruperi de pagină în formulare sau în rapoarte. În cazul formularelor utilitatea lor nu este apreciată, întrucât rare sunt situațiile în care un formular este format din mai multe pagini. Anteturile de pagină sunt folosite pentru a repeta datele care trebuie să apară în partea de sus a ecranului.

Redimensionarea acestor secțiuni poate fi realizată cu ajutorul mouse-ului prin agățarea marginilor acestora și mutarea într-o altă poziție.

### 3.3.5.2. Caseta cu instrumente

Toolbox

Caseta cu instrumente este elementul cel mai puternic de creare a diferitelor controale din cadrul unui formular. Activarea sau inhibarea ei atunci când editați un formular se face prin deschide-rea meniului "View" și activarea sau dezactivarea opțiunii "Toolbox". Butoanele "Select Objects" și "Control Wizards" nu plasează controale în formular. Butonul Select Objects, atunci când este apăsat vă permite selecția controalelor create în formulare, iar butonul Control Wizard, atunci când este apăsat permite intervenția asistenților Wizard pentru crearea anumitor controale. Butonul "More Controls" afișează o listă cu toate controalele Active X care sunt instalate în sistemul dumneavoastră.

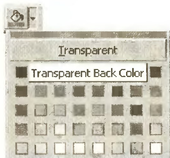
Butoanele “Rectangle” și “Line” vă permit introducerea unor elemente grafice simple (dreptunghiuri, linii) în formularul creat. Butonul “Page Break” permite forțarea începerii unei noi pagini într-un formular sau raport. Pentru a vedea explicațiile celorlalte controale din caseta cu instrumente revedeți tabelul de la punctul “Tipuri de controale” al acestui capitol.

Inserarea acestor controale în formularul creat se face prin apăsarea pe butonul corespunzător de pe caseta cu instrumente, urmată de revenirea în formular și trasarea cu ajutorul butonului stâng al mouse-ului a unui chenar în care va fi inserat controlul selectat.

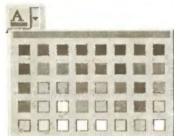
### 3.3.5.3. Paleta de culori



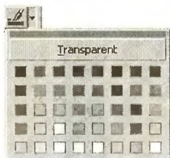
Funcționează doar în modul de editare a formularelor sau rapoartelor și permite, prin acțiunea combinată a mai multor controale, modificarea proprietăților de culoare sau de transparență ale obiectelor. Proprietatea implicită sau selectată de utilizator este afișată pe controlul corespunzător.




**Controlul Fill/Back Color** – Culoare de umplere – specifică ce culoare va fi folosită pentru fundal sau pentru controlul selectat.



**Controlul Font/Fore Color** – Culoarea fonturilor – stabilește culoarea fonturilor pentru controalele care folosesc text.




**Controlul Line/Border Color –** Culoarea de chenar – stabileşte liniei de contur a unui control din cadrul unui formular.



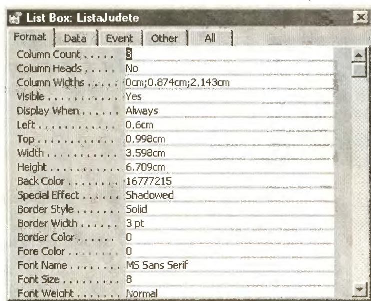
**Controlul Line/Border Width –** Stabileşte grosimea liniei de contur

**Controlul Special Efect** – Stabileşte efectul artistic 3D care va fi aplicat unui control.

Pentru aplicarea culorilor sau efectelor speciale unor controale din cadrul formularului curent, selectaţi controlul ale cărui proprietăţi doriţi să le modificaţi, deschideţi paleta de culori sau de efecte speciale şi executaţi un click pe culoarea sau efectul dorit.

### 3.3.5.4. Proprietăţile elementelor

În Access 97 fiecare control are proprietăţi specifice, care pot fi vizualizate şi modificate cu ajutorul casetei Properties. Afişarea casetei poate fi reali-zată prin exe-cutarea unui dublu click pe oricare dintre controalele formularului. Proprietăţile controalelor sunt grupate în patru categorii: Format, Data, Event, Other. Grupul All înglobează toate proprietăţile unui control aranjate în ordine alfabetică.




**Format** – încorporează toate proprietățile de prezentare a unui control: dimensiuni, poziția în cadrul formularului, culori, efecte, fonturi. Explicația şi rolul celor mai importante dintre acestea sunt:

| | |
|---|---|
| **Column Count** | Numărul de coloane ale controlului |
| **Column Heads** | Existența antetului de coloane |
| **Column Widths** | Lățimea coloanelor |
| **Visible** | Vizibilitatea în cadrul formularului. |
| **Left** | Distanța față de marginea stângă a formularului. |
| **Top** | Distanța. față de bara de titlu a formularului. |
| **Width** | Lățimea controlului. |
| **Back Color** | Codul numeric al culorii de fundal |
| **Special Efect** | Efectul 3D aplicat controlului. |
| **Font Color** | Culoarea fontului folosit de control. |
| **Font Name** | Numele fontului controlului. |

**Data** – înglobează toate proprietățile referitoare la sursa înregistrărilor care vor fi gestionate de un control.

| | |
|---|---|
| **Control Source** | Numele câmpului din tabelul sau interogarea sursă ale cărui valori vor afişate de acel control. |

**Row Source Type** Tipul sursei înregistrărilor Table/Query – tabel sau interogare – Value List – Valoare declarată de utilizator sau preluată ulterior din alte surse – Field List – numele câmpurilor din tabelul sau interogarea sursă.

**Row Source** Numele tabelului sau interogării sursă sau definiția în cod VBA a sursei înregistrării.

**Bound Column** Numărul implicite care conține datele care vor fi afişate dacă nu se fac referiri exprese la valorile altor coloane.

**Default Value** Valoarea implicită care va fi afişată în momentul activării acelui control.

**Enabled** Starea de activare a controlului. Setarea acestui control la valoarea No va bloca orice acțiune a utilizatorului asupra sa.

**Locked** Activează sau inhibă posibilitatea modificării datelor afişate de acel control.

**Event – Evenimente.** Conține lista tuturor evenimentelor care pot fi aplicate sau la care poate reacționa acel control. În câmpurile din partea dreaptă se vor menționa numele macroinstrucțiunilor sau procedurilor care vor fi executate la declanşarea acelui eveniment.

**Before Update** Înainte de modificarea valorii controlate.

**After Update** După modificarea valorii controlate.

**On Enter** La intrarea în acel control

**On Exit** La părăsirea controlului.

**On Got Focus** La captarea focusului de către control.

**On Lost Focus** La pierderea focusului de către control.

**On Click** La executarea unui click pe control.

**On Double Click** La executarea unui dublu click.

**On Mouse Down** În momentul apăsării nuia din butoanele mouse-ului.

| | |
|---|---|
| **On Mouse Move** | În momentul trecerii mouse-ului deasupra controlului. |
| **On Mouse Up** | La eliberarea butonului apăsat al mouse-ului. |
| **On Key Down** | La apăsarea unei taste. |
| **On Key Up** | La eliberarea tastei apăsate. |
| **On Key Press** | La apăsarea unei taste sau combinații de taste. |

**Other - Alte proprietăți.** Conține un set de proprietăți specifice ferestrelor Windows.

| | |
|---|---|
| **Pop Up** | Dacă este setată la valoarea "Yes" atunci formularul va fi menținut în prim planul ecranului chiar dacă la un moment dat nu el va fi formularul curent. |
| **Modal** | Atribuirea valorii "Yes" acestei proprietăți inhibă deschiderea altor formulare (sau efectuarea saltului la alte formulare) până la închiderea sa. |
| **Cycle** | Forțează deplasarea focusului numai între controalele înregistrărilor obiectului sursă (All Records), numai între controalele înregistrării curente (Current Record) sau numai între controalele înregistrărilor paginii curente (Current Page). |
| **Menu Bar** | Numele barei de meniu a utilizatorului care va fi afişată pe toată perioada lucrului în formularul curent. |
| **Toolbar** | Numele barei de unelte care va fi afişată pe ecran la deschiderea formularului. |
| **Shortcut Menu** | Valoarea "No" inhibă apariția meniului de context la executarea unui click cu butonul drept al mouse-ului. |

| | |
|---|---|
| **Shortcut Menu Bar** | Numele meiului de context care va afişat la executarea unui click cu butonul drept al mouse-ului. |
| **Help File** | Numele fişierului de help ataşat formularului, care va fi deschis la apăsarea tastei F1. |
| **Tag** | Conţine o valoare care poate fi memorată (şi folosită) împreună cu controlul căruia îi este atribuită. |
| **Name** | Numele controlului. |
| **Status Bar Text** | Expresia care fi afişată pe bara de status a formularului la activarea controlului. |
| **Control Tip Text** | Expresia care fi afişată pe ecran în momentul staţionării cursorului mouse-ului deasupra controlului. |
| **Tab Index** | Numărul de ordine al controlului în cadrul formularului. |

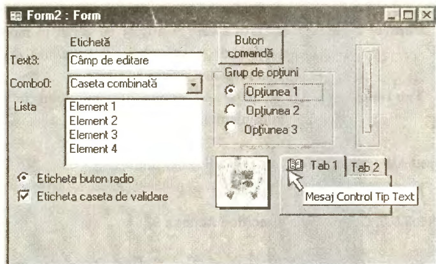
**Grupul All** - Conţine toate proprietăţile controlului.

### 3.3.5.5. Controalele specifice Access 97

Pentru a crea o interfaţă grafică (în cazul nostru un formular) care să satisfacă exigenţele majorităţii utilizatorilor este suficient să cunoaşteţi modul în care funcţionează (şi caracteristicile!) câtorva dintre controalele specifice sistemului de operare Windows. Aranjarea acestora în cadrul formularului, relaţiile dintre acestea precum şi comportamentul sau răspunsul la comenzile unui utilizator sunt întru totul la îndemâna dumneavoastră. Prin urmare, odată creată baza de date, nu vă mai rămâne decât să stabiliţi, de comun acord cu utilizatorii aplicaţiei create, modalitatea în care vor fi prezentate înregistrările pe ecran în vederea prelucrării.

**Etichetele** sunt casete de text transparente cu rol de transmite un anumit mesaj utilizatorului. Transparența acestora creează senzația "lipirii" textului de suprafața formularului. De regulă ele afişează numele unui control. Fără a afecta datele din tabelul sursă. Conținutul mesajului transmis de acestea nu poate fi modificat decât de acțiunea instrucțiunilor în cod VBA.

**Câmpurile de editare (Text Box)** sunt folosite pentru afişarea înregistrărilor. Utilizatorului îi este, de regulă, permisă intrarea în aceste câmpuri în vederea modificării sau introducerii datelor. Opțional acestor câmpuri li se ataşează o etichetă cu un mesaj explicativ.




**Casetele combinate (Combo Box)** sunt folosite pentru afişarea (pe una sau mai multe coloane) unui set de înregistrări selectate şi ordonate după un anumit criteriu. Sursa înregistrărilor o constituie, de regulă, o interogare sau coloanele tabelului sursă. Combo-urile sunt compuse dintr-un câmp de editare şi un buton de deschidere marcat de un triunghi cu vârful orientat în jos. Marele avantaj oferit de acest control (dat fiind că poate afişa seturi de sute sau mii de înregistrări) este economia de spațiu pe suprafața formularului, deoarece, în poziția închis, el nu afişează decât o singură valoare din întregul set de înregistrări pe care îl controlează. Deschiderea

acestui control poate fi realizată prin apăsarea tastei F4 sau prin executarea unui click pe butonul de deschidere. Opţional şi acestor controale li se poate ataşa o etichetă.

**Casetele cu listă (List Box)** îndeplinesc acelaşi rol ca şi casetele combinate, diferenţa constând în faptul că setul de înregistrări pe care îl controlează este afişat într-o casetă permanent deschisă. Pentru alegerea unui anumit element din setul de înregistrări afişat este suficientă executarea unui click pe acesta.

**Butoanele radio (Radio Button)** apar de regulă în grupuri de 2-3 şi sunt folosite pentru a "forţa" utilizatorul să aleagă o singură opţiune dintr-un set pus la dispoziţie de program. Alegerea uneia din opţiunile care se ascund în spatele acestor butoane radio se face prin executarea unui click fie pe butonul radio fie pe eticheta explicativă a butonului sau prin apăsarea combinaţiei de taste **ALT** + tasta subliniată (tastă scurtătură) din cadrul etichetei explicative. În funcţie de acţiunea utilizatorului butoanele radio pot lua două valori logice (True = Adevărat, când butonul este bifat şi False = Fals când butonul nu este bifat).

**Casetele de validare (Check Box)** permit alegerea, spre deosebire de butoanele radio) mai multor opţiuni din cadrul unui set pus la dispoziţie de program. Casetele de validare admit şi o a treia valoare **Null** atunci când caseta de validare se colorează în gri.

Din punct de vedere matematic valorile acceptate pentru casetele de validare sunt: -1 = pentru valoarea True, 0 = pentru valoarea False. Valoarea logică Null nu are corespondent matematic, de aceea recomandăm folosirea variabilelor logice pentru evitarea unor erori de prelucrare care ar putea genera bloca funcţionarea programului dumneavoastră.

**Butoanele de comandă (Command Button)** arată şi se comportă ca nişte butoane care pot fi apăsate pe un formular. La executarea unui click pe un astfel de buton, acesta dă impresia că este apăsat, după care revine la starea iniţială. Butoanele de comandă pot să conţină text sau imagini, pot să aibă orice dimensiune şi sunt folosite pentru lansarea în execuţie a unor secvenţe de cod VBA cu rezultate diferite în funcţionarea aplicaţiei.

**Eticheta de comutare (Tab Control)** oferă o metodă intuitivă de comutare între mai multe formulare. Este utilă în situația în care există foarte multe informații despre o singură entitate. Controalele Tab au proprietăți specifice și fiecare etichetă dintr-un control Tab are propria pagină de proprietăți. Dacă în modul de vizualizare Form se execută un click pe una din etichetele acestui control "pagina" etichetei corespunzătoare va fi afișată în prim plan, acoperind paginile celorlalte etichete.

**Grupul de opțiuni (OptionGroup)** Este cel mai indicat tip de control pentru a prezenta utilizatorului toate opțiunile disponibile., permițând selectarea unei singure opțiuni din întregul grup. În cazul utilizării unui grup de opțiuni acesta este "legat" de câmpul de date și apoi este completat cu unul sau mai multe dintre următoarele tipuri de controale:

- butoane de comandă
- butoane radio
- casete de validare

**Controalele Linie (Line) și Dreptunghi (Rectangle)** sunt folosite pentru înfrumusețarea formularelor

**Controalele Cadru de obiect neasociat (Unbound Object Frame) Cadru de obiect asociat (Bound Object Frame) și Imagine (Image).** Aceste trei controale sunt folosite pentru afișarea unor obiecte binare pe un formular. De obicei obiectele sunt documente realizate cu editoare de texte, imagini, sunete, foi de calcul tabelar, etc. Majoritatea programelor Windows pot să exporte obiecte cu ajutorul tehnicii OLE (Object Linked Embedded), iar programul Access 97 poate să afișeze astfel de obiecte și să folosească pentru editarea lor aplicația în care au fost create.

Obiectele neasociate (Unbound Object Frame și Image) nu se află în nici o relație cu datele înregistrate, astfel că modificările operate la nivelul bazei de date nu vor afecta aceste obiecte.

Obiectele asociate sau neasociate pot fi "legate" (Linked) sau înglobate (Embedded) într-un formular. Avantajul înglobării unui obiect într-un formular este creat de faptul că nu mai este necesară copia acelui

obiect pe discul calculatorului dumneavoastră. Un obiect legat stabileşte o referinţă în cadrul bazei de date cu obiectul însuşi, care trebuie să rămâne stocat într-un fişier de pe disc.

**Controlul Separator de Pagină (Page Break).** Dacă un formular este prea lung, puteţi să marcaţi un punct în care pagina pare că se rupe atunci când folosiţi tastele Page Up, Page Down. În cazul în care formularul este mai lung decât fereastra în care este afişat, Access 97 va afişa în mod automat o bară de derulare verticală pentru a permite navigarea în cadrul acestuia. Dacă este prezent un control Page Break tasta Page Down va redesena formularul astfel încât partea superioară a acestuia să apară imediat după separatorul de pagină. Acest tip de controale sunt extrem de utile în cazul rapoartelor.

### 3.3.5.6. Crearea unui formular

Formularul pe care ne propunem să-l realizăm va trebui să arate ca în figura următoare şi să îndeplinească următoarele sarcini:

Să afişeze lista judeţelor şi lista localităţilor unui judeţ în controale de tip casetă cu listă.

Să afişeze o casetă cu listă care să evidenţieze localităţile selectate de utilizator.

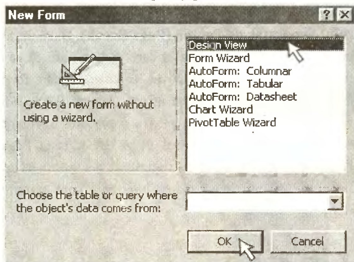
Să permită adăugarea, ştergerea, modificarea, sortarea şi tipărirea datelor înregistrate.

Instrucţiunile în cod VBA precum şi explicaţiile acestora vor fi oferite în finalul acestui subcapitol.

**Pasul 1:** Executaţi un click pe meniul Insert al aplicaţiei Access 97, urmat de un click pe opţiunea Form.

Executaţi un click pe opţiunea Design View, pentru a intra direct în modul de lucru proiectare, iar din caseta combinată care conţine numele tabelelor şi interogă-rilor bazei de date nu alegeţi nimic. Formularul pe care

îl vom crea în continuare este destul de complicat şi sursa înregistrărilor sale va fi definită mai târziu. În final apăsaţi pe butonul OK.




**Pasul 2:** În formularul nou creat, cu numele generic "Form1", nu va fi vizibilă decât secţiunea "Detail". Afişaţi antetul şi subsolul acestui formular executând un click pe meniul View al aplicației Access 97 şi un click pe opțiunea "Form Header/Footer".

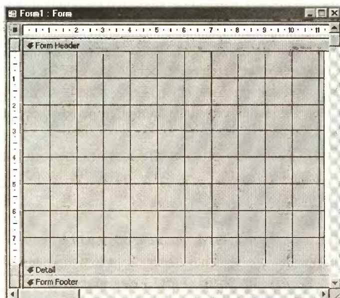
În cazul în care riglele orizontale şi verticale, rețeaua ortogonală şi bara de unelte nu sunt vizibile pe ecran la intrarea în modul de lucru proiectare, deschideţi meniul View şi bifaţi, pe rând, ca în figura alăturată, următoarele opţiuni: Ruler, Grid, Toolbox.




Redimensionarea secțiunilor formularului poate fi realizată prin agăţarea marginilor şi deplasarea acestora în direcţia dorită atunci când la cursorul mouse-ului apare una din următoarele forme: ⇕ + ✥ ⇔

În situația în care caseta de proprietăţi a unui control lipseşte din cadrul formularului executaţi un dublu click pe controlul ale cărui proprietăţi doriţi să le modificaţi.

Redimensionaţi secţiunile Form Header, Detail şi Form Footer astfel încât formularul dumneavoastră să arate ca în figura următoare.




Pentru a evita intervenția asistenților Wizard în activitatea de creare a controalelor formularului, asigurați-vă că butonul "Control Wizard" de pe bara de unelte nu este apăsat (în caz contrar executați un click pe el).

Dat fiind că este aproape imposibilă aprecierea dimensiunilor unui formular după o simplă fotografie, aşa cum apare în această lucrare, vom preciza valorile proprietăților formularului nostru care stabilesc dimensiunea şi poziția acestuia față de celelalte elemente ale ecranului.

Proprietățile din cadrul grupului Format al formularului:

**Rețineți:** expresiile scrise cu caractere înclinate reprezintă explicațiile efectelor proprietăților şi nu valorile acestora, deci nu le introduceți în câmpurile de editare.

**Width** 15.298 cm *înălțimea formularului*

**Min Max Buttons** None *nu vor fi afişate butoanele de minimizare şi maximizare pe bara de titlu a formularului.*

**Whats This Button** Yes *se afişează butonul de help senzitiv pe bara de titlu.*

Proprietățile din cadrul grupului Format al secțiunii Form Header:

**Height** 8.108 cm *lățimea zonei de antet (implicit şi a formularului).*

**Back Color** -2147483633 *culoarea de fundal a zonei de antet.*

Creați pentru început trei casete cu listă. Acest lucru îl puteți realiza apăsând butonul “List Box” de pe bara de unelte și trasând în interiorul secțiunii Form Header un dreptunghi care va reprezenta conturul casetei cu listă. Odată cu crearea controalelor aplicația Access 97 va crea și etichetele explicative ale casetelor cu listă (List, List2, List4) situație nedorită de această dată. Pentru a le șterge executați un click pe eticheta listei și apăsați tasta Delete.

Modificați în continuare proprietățile acestor casete cu listă după cum urmează:

Pentru prima listă:

Din grupul de proprietăți **Format**:

**Column Count** 3 *numărul de coloane afișate în interiorul listei.*

**Column Widths** 0cm;0.874cm;2.143cm *lățimea coloanelor.*

**Left** 0.6cm *dist. față de marginea din stânga a formularului.*

**Top** 0.998cm *dist. față de marginea superioară a formularului.*

**Width** 3.598cm *înălțimea casetei..*

**Height** 6.709cm *lățimea casetei.*

**Special Effect** Shadowed *caseta umbrită.*

**Border Width** 3 pt *grosimea în puncte a conturului casetei.*

Din grupul **Other**:

**Name** ListaJudete *numele castei cu listă.*

Proprietățile celei de-a doua liste diferă de proprietățile primei liste doar prin următoarele:

| | |
|---|---|
| **Left** | 4.298cm |
| **Top** | 0.998cm |
| **Width** | 2.725cm |
| **Height** | 6.698cm |
| **Column Count** | 4 |
| **Column Width** | 0cm;2.886cm;0cm;0cm |
| **Name** | ListaLocalitati |

Proprietățile celei de-a treia liste diferă de proprietățile primei liste prin următoarele:

| | |
|---|---|
| **Left** | 7.399cm |
| **Top** | 1.099cm |
| **Width** | 3.598cm |
| **Height** | 3.801cm |
| **Special Effect** | Sunken |
| **Column Count** | 1 |
| **Column Width** | 2.54cm |
| **Row Source Type** | Value List *Tipul sursei înregistrărilor.* |
| **Name** | ListaSelectii |

Creați în continuare trei butoane de comandă folosind tehnica anterioară (click pe butonul corespunzător controlului, pe care îl creați, de pe bara de unelte, revenire în zona de lucru a formularului în care va fi afişat controlul şi trasarea chenarului corespunzător conturului formularului). Aceste butoane le vom folosi ulterior pentru adăugări, ştergeri şi sortări de date.

După crearea acestor butoane modificați-le proprietățile după cum urmează:

***Pentru primul buton***:

| | |
|---|---|
| **Left** | 7.497cm |
| **Top** | 5.099cm |
| **Width** | 3.519cm |
| **Height** | 0.499cm |
| **Caption** | Adaugă *Expresia afişată pe suprafața butonului.* |
| **Name** | BtnAdauga |

***Pentru cel de-al doilea buton:***

| | |
|---|---|
| **Left** | 7.497cm |
| **Top** | 5.598cm |
| **Width** | 3.519cm |
| **Height** | 0.631cm |
| **Caption** | Şterge |
| **Name** | BtnSterge |

*Pentru cel de-al treilea buton:*

| | |
|---|---|
| **Left** | 7.497cm |
| **Top** | 6.198cm |
| **Width** | 3.519cm |
| **Height** | 0.631cm |
| **Caption** | Sortează |
| **Name** | BtnSorteaza |

Creați în continuare o ***casetă combinată*** cu următoarele proprietăți:

| | |
|---|---|
| **Left** | 7.497cm |
| **Top** | 6.698cm |
| **Width** | 3.501cm |
| **Height** | 0.409cm |
| **Row Source Type** | Value List |
| **Row Source** | Ascendent |
| **Name** | CboTipsortare |

În sfârşit ultimul control pe care trebuie să-l creați pentru a obține un formular identic, cel puțin ca formă, cu cel prezentat de noi, este un ***buton de comandă*** cu următoarele proprietăți:

| | |
|---|---|
| **Left** | 7.497cm |
| **Top** | 7.099cm |
| **Width** | 3.519cm |
| **Height** | 0.631cm |
| **Caption** | Tipăreşte |
| **Name** | BtnTipareste |

**Pasul 3:** Pentru a salva formularul creat, deschideți meniul file al aplicației Access 97 şi alegeți opțiunea Save din acesta. Veți primi imediat un mesaj de avertizare asemănător cu cel din figura alăturată, în care trebuie să precizați numele formularului pe care l-ați creat. Denumiți-l deocamdată Form1, aşa cum ne propune şi figura. După salvarea formularului, închideți-l pentru a putea trece la faza următoare a creării unui formular – adăugarea codului.



# CODUL VBA

## Noțiuni generale

Scrierea codului pe ecran presupune organizarea clară a ideilor, înțelegerea posibilităților și limitărilor mediului și cunoașterea celor mai importante elemente referitoare la sintaxă.

În acest capitol ne propunem crearea unor secvențe de cod care să trateze evenimente referitoare la acțiunea butoanelor de comandă și a controalelor formularelor. Vor fi folosite secvențe de instrucțiuni destul de dificile cu scopul de a vă familiariza cu modalitatea de combinare a diferitelor secvențe de cod pentru realizarea scopurilor propuse.

Principalul avantaj al programării în cod VBA este standardizarea pe care o asigură întregului pachet de aplicații Microsoft Office (Word, Excel, Power Point, Outlook).

## 4.1. Proceduri

Întregul cod VBA este scris în ***proceduri***, care sunt blocuri de instrucțiuni cu început și sfârșit bine determinat. În cadrul modulelor procedurile nu pot fi imbricate, însă, în general, ele pot apela alte proceduri, iar un cod bine scris constă din multe proceduri scurte, ușor de înțeles, care se apelează între ele. Acest tip de programare poartă numele de programare modulară, iar codul este ușor de înțeles și de întreținut.

În limbajul VBA există două tipuri de proceduri:

***Subrutinele*** sunt proceduri care nu returnează nici o valoare. Ele au un început bine definit (o instrucțiune Sub) și un sfârșit bine definit (o

instrucțiune End Sub). Subrutinele pot avea parametri, incluzând numele și tipul parametrului pe prima linie a subrutinei, ca în exemplul următor:

Sub Calculează (Număr As String)

***Funcțiile*** sunt proceduri care returnează o valoare procedurii care le-a apelat. Ele au un început bine definit (o instrucțiune Function) și un sfârșit bine definit (o instrucțiune End Function). Prima linie a unei funcții definește tipul de dată al valorii pe care o va returna. Funcțiile pot avea parametri, incluzând numele și tipul parametrului pe prima linie a funcției, ca în exemplul următor:

Function Calculează (Număr As Long) As Long.

## 4.2. Instrucțiuni în cod VBA

Instrucțiunile în cod VBA reprezintă baza funcționării programelor create de utilizatori de aceea este extrem de importantă cunoașterea acestora și a sintaxei lor. În continuare vom prezenta instrucțiunile cele mai des folosite la realizarea programelor în Access 97.

### 4.2.1. Funcții pentru conversia datelor

**Asc** returnează un număr întreg reprezentând codul ANSI al primului caracter din șirul analizat.

Sintaxa comenzii este: **Asc(*șir*)**

**CCur** convertește o expresie într-o valoare de tip Currency, așa cum este ea definită de sistemul de operare.

Sintaxa comenzii este: **Ccur(*expresie*)**

**CDate** convertește o expresie într-o valoare de tip dată calendaristică.

Sintaxa comenzii este: **CDate(*expresie*)**

**CDbl** convertește o expresie într-o valoare de tip Double.

Sintaxa comenzii este: **CDbl(*expresie*)**

**CInt** convertește o expresie într-o valoare de tip Integer.

Sintaxa comenzii este: **CInt(*expresie*)**

**Chr** returnează un caracterul asociat al unui număr în cod ANSI.

Sintaxa comenzii este **Chr(*număr*).**

**Format** returnează un şir formatat în concordanţă cu instrucţiunile de formatare.

Sintaxa comenzii este: **Format(expresie[, format])**

**Val** returnează valoarea unui număr conţinut într-un şir de tip text.

Sintaxa comenzii este: **Val(şir).**

### 4.2.2. Funcţii pentru controlul fluxului

**For ... Next** - incrementează un contor.

Sintaxa comenzii este:

**For** *contor = val de start* **To** *val sfârşit* **[Step** *pas***]**

[*comenzi VBA*]

**[Exit For]**

[*comenzi VBA*]

**Next** [*contor*]

Funcţia **For...Next** are următoarele părţi:

***contor*** Obligatorie. Variabilă numerică folosită pentru contorizare. Poate fi de tip logic sau de tip şir.

***Val de start*** *Obligatorie*. Valoarea iniţială a contorului.

***Val sfârşit*** *Obligatorie*. Valoarea finală a contorului.

***pas*** *Opţional*. Valoarea de incrementare a contorului la fiecare parcurgere a ciclului. Dacă nu este specificată se presupune că este egală cu 1.

***comenzi*** *Opţional*. Una sau mai multe comenzi VBA incluse în ciclul For ... Next care vor fi executate de un număr de ori.

**For Each ... Next** - Repetă un grup de instrucţiuni pentru fiecare element al unui şir sau al unei colecţii.

Sintaxa comenzii este:

**For Each** *element* **In** *grup*
*[comenzi]*
**[Exit For]**
[*comenzi*]
**Next** [*element*]

Unde:

***element*** *Obligatoriu.* Reprezintă elementele unui grup sau colecții care se analizează.

***grup*** *Obligatoriu.* Numele grupului din care fac parte obiectele de analizat.

**If ... Then ... Else -** Execută condiționat de valoarea unei expresii un grup de comenzi VBA.

Sintaxa comenzii este:

**If** *condiție* **Then**
[*comenzi*]
**[ElseIf** *condiție - n* **Then**
*[comenzi]* **...**
**[Else**
[*comenzi*]**]**
**End If**

**Select Case** - Execută condiționat de valoarea unei expresii un grup de comenzi VBA.

Sintaxa comenzii este:

**Select Case** *condiție*
**[Case** *condiție - n*
  [*comenzi*]**] ...**
**[Case Else**
  [*comenzi*]**]**
**End Select**

**Do...Loop** - Asigură executarea unui set de comenzi VBA atât timp cât este îndeplinită o anumită condiție, sau până la îndeplinirea unei condiții.

Sintaxa acestei comenzi este:

**Do [{While | Until}** *condiție îndeplinită*]
[*comenzi*]
**[Exit Do]**
[*comenzi*]
**Loop**

Este permisă și folosirea unei sintaxe de tipul:

**Do**
[*comenzi*]
**[Exit Do]**
[*comenzi*]
**Loop [{While | Until}** *condiție îndeplinită*]

**On Error** - Activează sau inhibă acțiunea unei rutine de tratare a erorilor.

Această comandă poate avea una din următoarele sintaxe:

**On Error GoTo** numărul liniei de cod

În cazul apariției unei erori se execută un salt la linia de program specificată.

**On Error Resume Next**

Se execută un salt necondiționat la linia de program imediat următoare celei care a generat eroarea.

**On Error GoTo 0**

Dezactivează orice rutină de tratare a erorilor.

**End** - Asemănătoare instrucțiunii End Function sau End Sub.

**Exit** - Folosită ca "Exit Function" sau "Exit Sub".

### 4.2.3. Funcții matematice

**Abs**(*număr*) Returnează valoarea absolută a unui număr. Dacă numărul conține valoarea Null, funcția va returna valoarea Null, iar dacă argumentul funcției nu a fost inițializat va fi returnată valoarea 0.

Exemplu:

```
Dim Număr
Număr = Abs(50.3)     ' Returnează 50.3.
Număr = Abs(-50.3)    ' Returnează 50.3.
```

**Atn**(*număr*) Returnează o valoare de tip Double, reprezentând arctangenta acelui număr exprimată în radiani. Argumentul acestei funcții poate fin un număr de tip Double sau orice expresie numerică.

Exemplu:

```
Dim pi
pi = 4 * Atn(1)  ' Calculul valorii numărului pi.
```

**Cos**(*număr*) Returnează o valoare de tip Double, reprezentând cosinusul unui unghi exprimat în radiani.

Exemplu:

```
Dim Unghi, Secantă
Unghi = 1.3                ' Defineşte unghiul în radiani.
Secantă = 1 / Cos(Unghi)   ' Calculează secanta unghiului.
```

**Exp**(*număr*). Returnează o valoare de tip Double, reprezentând valoarea numărului e (baza logaritmilor naturali) ridicată la o putere. Dacă argumentul acestei funcții este mai mare de 709,782712893 va fi generată o eroare de calcul. Valoarea numărului e este aproximată la 2,718282.

Exemplu:

```
Dim Unghi, SinusHip
```

Unghi = 1.3 ' Defineşte unghiul în radiani.

SinusHip = (Exp(Unghi) - Exp(-1 * Unghi)) / 2' Calculează sinusul hiperbolic.

**Int(*număr*)**. Returnează partea întreagă a unui număr.

Exemplu:

Dim Număr

Număr = Int(99.8) ' Returnează 99.

Număr = Int(-99.8) ' Returnează -100.

**Log(*număr*)**. Returnează valoarea logaritmului natural al unui număr

Exemplu:

Log *n(x)* = Log*(x)* / Log*(n)* 'Formula pentru calculul logaritmului în baza *n* al unui număr *x*.

Dim Unghi, Logaritm

Unghi = 1.3 ' Definesc unghiul în radiani.

Logaritm = Log(Unghi + Sqr(Unghi * Unghi + 1)) ' Calculez inversul sinusului hiperbolic.

**Rnd(*număr*)** Returnează o valoarea aleatoare de tip Single. Argumentul funcției trebuie să fie o număr de tip Single sau orice expresie numerică.

Exemplu.

Dim NumărAleator

NumărAleator = Int((6 * Rnd) + 1) 'Generez o valoarea aleatoare cuprinsă între 1 şi 6.

**Sin(*număr*)** Returnează o valoare de tip Double, reprezentând sinusul unui unghi exprimat în radiani.

Exemplu:

Dim Unghi, Cosecanta

Unghi = 1.3 ' Definesc unghiul în radiani.

Cosecanta = 1 / Sin(Unghi) ' Calculez cosecanta unghiului.

**Sqr(***număr***)**. Returnează radicalul de ordinul doi al unui număr.

Exemplu:
Dim Radical
Radical = Sqr(23) ' Returnează 4.79583152331272.
Argumentul funcției trebuie să fie un număr de tip Double.

### 4.2.4. Funcții pentru declararea variabilelor

Aceste funcții sunt:

| **Funcția** | **Tipul Datelor** |
|---|---|
| **DefBool** *litera*[*,litera*] . . . | Boolean |
| **DefByte** *litera*[, *litera*] . . . | Byte |
| **DefInt** *litera*[, *litera*] . . . | Integer |
| **DefLng** *litera*[, *litera*] . . . | Long |
| **DefCur** *litera*[, *litera*] . . . | Currency |
| **DefSng** *litera*[, *litera*] . . . | Single |
| **DefDbl** *litera*[, *litera*] . . . | Double |
| **DefDec** *litera*[, *litera*] . . . | Decimal |
| **DefDate** *litera*[, *litera*] . . . | Date |
| **DefStr** *litera*[, *litera*] . . . | String |
| **DefObj** *litera*[, *litera*] . . . | Obiect |
| **DefVar** *litera*[, *litera*] . . . | Variant |

Sunt folosite la nivelul modulelor pentru declararea tipului de date al variabilelor, argumentelor procedurilor și pentru returnarea tipului funcțiilor sau procedurilor al căror nume începe cu litera specificată. Aceste funcții au efect doar în interiorul modulelor în care au fost introduse.

Exemplu:

**DefInt A-K** 'Impun tipul Integer pentru toate variabilele al căror nume începe cu una din literele de la A la K.

**DefStr L-P** 'Impun tipul String pentru toate variabilele al căror nume începe cu una din literele de la L la P.

**DefDbl T** 'Impun tipul Double pentru toate variabilele al căror nume începe cu litera T (sau t).

Comanda **Dim -** Este folosită pentru declararea explicită a unor variabile şi alocarea spaţiului necesar acestora.

Sintaxa comenzii:

**Dim [WithEvents]** *numevar*[([*dimensiune*])] **[As [New]** *tip*] [, **[WithEvents]** *numevar*[([*dimensiune*])] **[As [New]** *tip*]] . . .

Expresiile din paranteze drepte au caracter opţional, depinzând de contextul în care sunt folosite, iar argumentele funcţiei au următoarele semnificaţii:

**WithEvents** Cuvânt sau simbol recunoscut de VBA care specifică faptul că numevar este o variabilă care conţine referinţe către un obiect, folosit pentru a răspunde evenimentelor ataşate obiectelor altor aplicaţii externe.

*numevar* Numele variabilei.

*dimensiune* Dimensiunea unei variabile de tip şir (max. 60). Argumentul dimensiunii are următoarea sintaxă:

*Minim* **To** *maxim.* Implicit *Minim* are valoarea 0.

**New** Cuvânt sau simbol recunoscut de VBA care activează crearea implicită a unui obiect nou.

*tip* Tipul variabilei.

Exemplu:

**Dim** AnyValue, MyValue ' AnyValue şi MyValue sunt declarate ca fiind de tip Variant cu valoare implicită Null.

**Dim** Number As Integer ' Declarație explicită a unei variabile de tip Integer..

**Dim** AnotherVar, Choice **As** Boolean, BirthDate **As** Date

' Declarații multiple pe o singură linie. AnotherVar este de tip Variant deoarece a fost omis tipul acesteia în declarație.

**Dim** DayArray(50) ' DayArray este un şir de tip Variants cu 51 elemente indexate, de la 0 la 50, cu valoarea minimă implicită 0.

**Dim** Matrix(3, 4) **As** Integer ' Matrix este un şir bidimensional de tip Integer.

**Dim** MyMatrix(1 To 5, 4 To 9, 3 To 5) **As** Double ' MyMatrix este un şir tridimensional de tip Double cu *Minim* şi *Maxim* explicit.

**Dim** BirthDay(1 To 10) **As** Date' BirthDay este un şir de date calendaristice indexate de la 1 la 10.

**ReDim** - folosită la nivelul procedurilor pentru realocarea spațiului necesar variabilelor.

Sintaxa comenzii:

**ReDim [Preserve]** *varname*(*dimensiune*) **[As** *type*] ...

Unde:

**Preserve** Optional. Cuvânt sau simbol recunoscut de VBA folosit pentru păstrarea datelor unui şir declarat la schimbarea ultimei dimensiuni.

*varname* numele variabilei.

*dimensiune* Dimensiunea unei variabile de tip şir (max. 60).

*type* tipul variabilelor..

**Set -** Atribuie proprietățile unui obiect unei variabile sau proprietăților unui alt obiect.

Sintaxa comenzii este următoarea:

**Set** *objectvar* = {[**New**] *objectexpression* | **Nothing**}

Unde:

*objectvar* Numele proprietății sau variabilei..

**New** Cuvânt sau simbol recunoscut de VBA care activează crearea implicită a unui obiect nou.

*objectexpression* numele obiectului, variabilei, proprietății unui alt obiect căruia i se atribuie o proprietate.

**Nothing** Anulează efectul de asociere al *objectvar* cu obiectul specificat.

Exemplu:

```
Dim YourObject, MyObject, MyStr
Set MyObject = YourObject           ' Atribuim proprietățile obiectului
                                    YourObject, obiectului MyObject..
YourObject.Text = "Hello World"     ' Atribui expresia "Hello World"
                                    proprietății Text a obiectului
                                    YourObject.
MyStr = MyObject.Text               ' Atribuim explicit (doar) proprietatea
                                    Text a obiectului MyObject.
Set MyObject = Nothing              ' Întrerup viitoarele referiri la propri-
                                    etățile obiectului sursă YourObject..
```

**Public** - folosită la nivelul modulelor pentru declararea variabilelor și alocarea spațiului necesar acestora.

Sintaxa comenzii:

**Public [WithEvents]** numevar[([dimensiune])] **[As [New]** tip]. . .

**Private** - folosită la nivelul modulelor pentru declararea variabilelor specifice unui modul sau unei proceduri și pentru alocarea spațiului necesar acestora.

Sintaxa comenzii:

**Private [WithEvents]** numevar[([dimensiune])] **[As [New]** tip]. . .

Spre deosebire de variabilele declarate cu ajutorul comenzii **Public**, cele declarate cu ajutorul acestei instrucțiuni au valabilitate şi există doar în interiorul modulului sau procedurii în care au fost declarate.

**Static** - folosite la nivel de proceduri pentru declararea variabilelor şi alocarea spațiului necesar acestora. Variabilele declarate cu ajutorul acestei comenzi îşi păstrează valorile între două apeluri ale procedurii care le foloseşte.

Sintaxa comenzii:

**Static** varname**[(**[subscripts]**)]** **[As [New]** type**]**...

**Const** - folosite pentru declararea constantelor care vor fi folosite în locul variabilelor literare.

Sintaxa comenzii:

**[Public | Private] Const** numeconstantă **[As** tip**]** = expresie

### 4.2.5. Funcții pentru controlul datei

**Date** - Returnează o valoare de tip Variant reprezentând data curentă a sistemului.

Sintaxa comenzii:

**Date** = Data

Exemplu:

**Dim** MyDate

MyDate = #February 12, 2001#  ' Stabilesc noua valoare a datei.

**Date** = MyDate ' Schimbăm data sistemului

**Now** - Returnează o variabilă de tip Variant reprezentând data şi ora curentă a sistemului.

Exemplu:

MyDate = **Now()**

**Time** - Returnează o variabilă de tip Variant reprezentând valoarea orei curente a sistemului.

Exemplu:

MyTime1 = **Time()** 'Determin ora curentă a sistemului.

MyTime2 = **Time()** + (1/24)

### 4.2.6. Funcții pentru returnarea şi prelucrarea şirurilor

Aceste funcții returnează fie o valoare de tip Variant fie una de tip String.

**Chr$** - Returnează o variabilă de tip String conținând caracterul asociat codului ANSI menționat.

Exemplu:

**Dim** MyChar

MyChar = **Chr**(65) ' Returnează A.

MyChar = **Chr**(97) ' Returnează a.

MyChar = **Chr**(62) ' Returnează >.

MyChar = **Chr**(37) ' Returnează %.

**CurDir$** - Returnează o variabilă de tip Variant reprezentând calea curentă.

Exemplu:

Presupunem că unitatea curentă este "C:", directorul curent este "C:\Windows", şi calea curentă spre o altă unitate este: "D:\My Documents"

**Dim** MyPath

MyPath = **CurDir** ' Returnează "C:\ ".

MyPath = **CurDir**("C") ' Returnează "C:\Windows ".

MyPath = **CurDir**("D") ' Returnează "D:\My Documents".

**Dir$** - Returnează o variabilă de tip String reprezentând numele unui fişierelor sau folderelor care au calea şi atributele specificate de utilizator.

Sintaxa comenzii:

**Dir**[(cale[, atribute])]

cale *Opțional.* O expresie de tip String reprezentând numele unui fişier, folder sau al unei unități de memorie. Dacă nu este specificată nici o cale de căutare va fi returnat un şir de lungime zero.

atribute *Opțional.* Constantă sau expresie numerică care însumează atributele specificate.

Denumirile constantelor şi valorile acestora:

| Denumire | Valoare | Descriere |
|---|---|---|
| vbNormal | 0 | Normal |
| vbHidden | 2 | Ascuns |
| vbSystem | 4 | Fişier de sistem |
| vbVolume | 8 | Eticheta de volum a unei unități de memorie. |
| vbDirectory | 16 | Folder sau director. |

Exemplu:

**Dim** Fişier, Cale, Nume

Fişier = **Dir**("C:\WINDOWS\WIN.INI") 'Returnează numele fişierului WIN.INI dacă acesta există.

Fişier = **Dir**("C:\WINDOWS*.INI") 'Returnează numele primului fişier găsit în directorul WINDOWS cu extensia INI.

Fişier = **Dir** 'Repetând comanda va fi returnat numele următorului fişier cu extensia INI.

Fişier = **Dir**("*.TXT", vbHidden) ' Va fi returnat primul fişier găsit care are extensia TXT şi atributul Hidden.

Pentru a afişa lista tuturor directoarelor din directorul rădăcină al unei unităţi de memorie recomandăm următoarea secvenţă de instrucţiuni:

Cale = "c:\" ' Stabilesc calea implicită.

Nume = **Dir**(Cale, vbDirectory) ' Preiau numele primului director din rădăcina unităţii C:.

**Do While** Nume <> ""

' Deschid un ciclu de execuţie care va dura atâta timp cât variabila Nume va avea o .valoare diferită de Null.

**If** Nume <> "." **And** Nume <> ".." **Then**

' Dacă Nume este diferită de . şi .. atunci:

**If** (GetAttr(Cale & Nume) And vbDirectory) = vbDirectory Then

'Dacă obiectul obţinut din compunerea căii şi a numelui este un director atunci:

**Debug.Print** Nume ' Tipăreşte numele directorului.

**End If** ' it represents a directory.

**End If**

Nume = **Dir** ' Atribui nouă valoare variabilei Nume..

**Loop** ' Trecem la următorul director din rădăcină.

**Error$** Returnează un mesaj de eroare care corespunde unui număr de cod de eroare.

Sintaxa comenzii este:

**Error**(număr_eroare) 'unde număr_eroare poate fi orice număr cuprins între şi 65535.

Exemplu:

Dim Număr_Eroare

**For** Număr_Eroare = 61 To 64

**Debug.Print Error**(Număr_Eroare) ' Tipăresc numărul erorii.

**Next** Număr_Eroare

**Hex$** - Returnează o variabilă de tip String reprezentând valoarea unui număr în sistemul de numerație hexazecimal.

Sintaxa comenzii: **Hex$**(număr).

**Input$**- Returnează o variabilă de tip String conținând caracterele unui fișier deschis în modul Input sau binar.

Sintaxa comenzii:

**Input**(număr, [#]numărfișier)

Unde:

| | |
|---|---|
| număr | *Obligatoriu.* Orice expresie semnificând numărul caracterelor returnate. |
| numărfișier | *Obligatoriu.* Orice număr de fișier. |

Exemplu:

| | |
|---|---|
| **Dim** MyChar | |
| **Open** "TESTFILE" **For Input As** #1 | ' Deschid fișierul TESTFILE pentru preluare date acordându-I numărul generic 1. |
| **Do While Not EOF**(1) | ' Lansez un ciclu de execuție care va fi încheiat în momentul ajungerii la sfârșitul fișierului. |
| MyChar = **Input**(1, #1) | ' Preiau primul caracter din fișierul deschis în variabila MyChar. |
| **Debug.Print** MyChar | ' Tipăresc valoarea preluată în MyChar. |
| **Loop** | ' Reluăm operațiunea de preluarea a conținutului fișierului. |
| **Close** #1 | ' Închid fișierul la preluarea ultimului caracter.. |

**LCase$** - Transformă caracterele unei expresii de tip String în litere mici.

Sintaxa comenzii:

**LCase**(expresie)

Exemplu:

Dim Majuscule, Minuscule
Majuscule = "Hello World 1234" ' Definesc şirul care va fi convertit..
Minuscule = **LCase**(UpperCase) ' Convertesc conţinutul variabilei Majuscule în litere mici şi-l atribui variabilei Minuscule, astfel că la o eventuală apelare a acesteia va fi returnat şirul "hello world 1234".

**Left$** - Returnează o valoare de tip Variant sau String , conţinând un număr specificat de caractere din partea stângă a şirului analizat.

Sintaxa comenzii:

**Left**(şir, lungime)

Exemplu:

**Dim** Expresie, Şir
Expresie = "Hello World" ' Definesc şirul.
Şir = Left(Expresie, 1) ' Returnează "H".
Şir = Left(Expresie, 7) ' Returnează "Hello W".
Şir = Left(Expresie, 20) ' Returnează "Hello World".

**LTrim$** - Returnează un şir de tip Variant sau String format din caracterele şirului original cu excepţia spaţiilor incluse în partea stângă a acestuia.

Sintaxa comenzii:

**Ltrim**(Şir)

Exemplu:

**Dim** ŞirPrelucrat, Şir
Şir = " <-Trim-> " ' Iniţializez şirul.
ŞirPrelucrat = **LTrim**(Şir) ' Returnează "<-Trim-> ".

**Mid$** - Returnează un şir de tip Variant sau String format dintr-un număr specificat de caractere ale şirului inițial.

Sintaxa comenzii:

**Mid**(şir, start[, lungime])

Exemplu:

**Dim** şir, Primul, Ultimul, Mijloc
şir = "Mid Function Demo" ' Definesc şirul.
Primul = **Mid**(şir, 1, 3) ' Returnează "Mid".
Ultimul = **Mid**(şir, 14, 4) ' Returnează "Demo".
Mijloc = **Mid**(şir, 5) ' Returnează "Function Demo".

**Right$**- Returnează o valoare de tip Variant sau String, conținând un număr specificat de caractere din partea dreaptă a şirului analizat. Funcționarea este asemănătoare comenzii **Left** cu excepția faptului că numărarea caracterelor începe din partea dreaptă a şirului inițial.

**RTrim$**- Returnează un şir de tip Variant sau String format din caracterele şirului original cu excepția spațiilor incluse în partea dreaptă a acestuia.

**Space$** - Permite introducerea unui număr specificat de spații.

Sintaxa comenzii:

**Space**(număr)

Exemplu:

**Dim** Şir
Şir = Space(10) 'Definesc un şir format din 10 spații
Şir = "Hello" & **Space**(10) & "World" ' Inserez cele 10 spații între două şiruri.
**Str**(număr) 'Converteşte o valoare de tip numeric într-una de tip şir.

**Trim$** 'Returnează un şir format din caracterele şirului original fără spaţii.

**Ucase** Converteşte în majuscule caracterele unui şir. V. **Lcase**.

### 4.2.7. Funcţii uzuale

**MsgBox** - Afişează un mesaj într-o casetă de dialog, aşteaptă un răspuns din partea utilizatorului şi returnează valoare de tip Integer indicând butonul care a fost apăsat.

Sintaxa comenzii:

**MsgBox**(mesaj[, buton] [, titlu] [, fişier_de_help, context])

mesaj *Obligatoriu.* Expresia care fi afişată în caseta de dialog. Lungimea maximă a mesajului este de 1024 caractere. Dacă mesajul are mai multe linii, acestea pot fi separate apelând codul ASCII al tastei Enter (Chr(13)) sau al sfârşitului de linie (Chr(10)).

buton *Opţional.* Expresie numerică reprezentând suma valorilor numărului de butoane şi tipului acestora, al iconiţei folosite, valorii butonului implicit şi tipului de casetă de mesaj. Implicit este 0.

titlu *Opţional.* Expresia afişată pe bara de titlu a casetei de mesaj. Implicit este dat de numele aplicaţiei.

fişier_de_help *Opţional.* Numele fişierului de help care va fi deschis la solicitarea referinţelor unui obiect.

context *Opţional.* Expresie numerică pentru identificarea indexului fişierului de help.

Valorile returnate de butoanele casetei de mesaj:

| Constantă | Valoare | Descriere buton |
|---|---|---|
| vbOK | 1 | OK |
| vbCancel | 2 | Cancel |
| vbAbort | 3 | Abort |

| | | |
|---|---|---|
| vbRetry | 4 | Retry |
| vbIgnore | 5 | Ignore |
| vbYes | 6 | Yes |
| vbNo | 7 | No |

Exemplu:

**Dim** mesaj, titlu

Titlu = "Costinești ..." ‘Definesc conținutul titlului casetei de mesaj

Mesaj = "Stațiunea vacanțelor!" ‘Definesc conținutul mesajului care va fi afişat.

**MsgBox** Mesaj, vbOkOnly, Titlu ‘Caseta va conține mesajul, titlul şi un buton OK.

**InputBox** - Afişează o casetă de mesaj, aşteaptă răspunsul utilizatorului şi returnează un şir introdus de utilizator.

Sintaxa comenzii:

**InputBox**(mesaj[,titlu] [,text_implicit] [,Xpos] [,Ypos] [,fişier_de_help, context])

Unde

Xpos reprezintă distanța pe orizontală a casetei de mesaj față de marginea din stânga a formularului principal.

Ypos reprezintă distanța pe verticală față de marginea superioară a formularului principal.

Exemplu:

**Dim** Şir **As String**, Mesaj **As String**

Mesaj = "Enter your name." ‘Definesc mesajul care va fi afişat.

Şir = **InputBox**("Şir", "titlu", 2000, 2000) ‘Preiau textul care va fi introdus de utilizator în variabila Şir.

**MsgBox** "Hello, " & Şir ‘Afişez expresia Hello şi textul introdus de utilizator.

**IsNull** - Returnează o valoare de tip Boolean indicând faptul că variabila respectivă are o valoare validă.

Sintaxa comenzii:

**IsNull**(expresie)

Exemplu:

**Dim** Mesaj1, Mesaj2, Variabilă 'definesc variabilele necesare

Mesaj1 = "Variabilă nulă!"

Mesaj2 = "Variabilă definită!"

Variabilă = **InputBox** ("Introduceți o valoare!") 'Las utilizatorului posibilitatea de a defini valoarea variabilei Variabilă

**If IsNull**(Variabilă) **Then**

**MsgBox** Mesaj1 'Dacă utilizatorul nu a introdus nimic în câmpul de editare al casetei de dialog anterioare sau dacă a apăsat pe butonul Cancel va fi afişat conținutul variabilei Mesaj1

**Else**

**MsgBox** Mesaj2 'În caz contrar va fi afişat conținutul variabilei Mesaj2

**End If**

**Len** - Returnează o valoare de tip Long reprezentând numărul de caractere al unei expresii.

Sintaxa comenzii:

**Len**(expresie)

Exemplu:

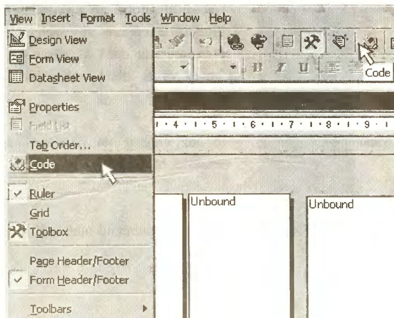
**MsgBox** Mesaj2 & " conține " & **Len**(Mesaj2) & " caractere"

'Afişez o casetă de mesaj care va conține. expresia: "Variabilă definită! conține 20 caractere"

## 4.3. Adăugarea codului

Adăugarea codului VBA, controalelor unui formular, este secțiunea cea mai importantă a programării în VBA, fără de care formularele aplicației dumneavoastră nu vor satisface niciodată pe deplin cerințele utilizatorilor. Pentru adăugarea codului va trebui să cunoașteți semnificațiile instrucțiunilor folosite care însă face obiectul capitolului 4. În această secțiune vom prezenta doar modalitățile de adăugare sau de modificare a codului ascuns în spatele diferitelor controale.

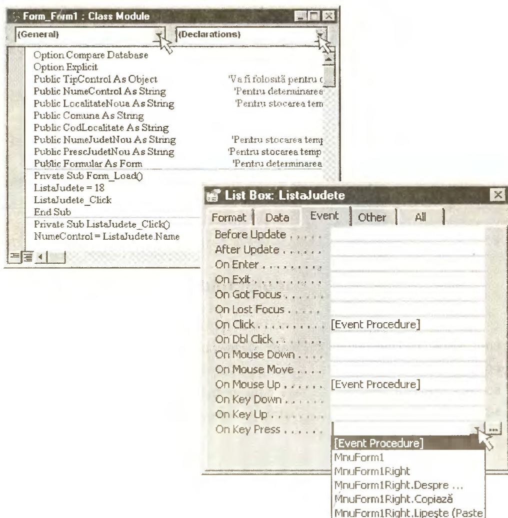
Deschideți formularul Form1 pe care l-ați creat mai devreme în modul de lucru Design și executați un dublu click pe unul din controalele formularului pentru a vă asigura prezența pe ecran a casetei de proprietăți. Pentru a fi în ton cu exemplul dat de noi .




În figura alăturată se observă ușor că nu este neapărat necesară prezența ferestrei de proprietăți, vizualizarea codului putând fi efectuată fie prin alegerea opțiunii Code din meniul View, fie apăsarea pe butonul Code de pe bara de unelte Standard.

Fereastra care va apare pe ecranul calculatorului, în urmă executării oricăreia dintre cele 2 variante propuse anterior va conține:

O listă combinată de tip combo (expresia implicită "General") care înglobează numele obiectelor definite în cadrul formularului curent, o listă combinată cu numele evenimentelor atașate unuia dintre obiectele create în formularul curent sau numele procedurilor definite de utilizator și un număr oarecare de zone albe, separate printr-o linie orizontală, pentru introducerea sau modificarea instrucțiunilor VBA ale unei proceduri.



Varianta recomandată pentru deschiderea paginii de cod VBA, direct la procedura unui anumit obiect este următoarea:

**Pasul 1:** Deschideți caseta de proprietăți a unui control la pagina “Event” – Evenimente

**Pasul 2:** Selectați opțiunea “Event Procedure” în dreptul evenimentului necesar și executați

**Pasul 3:** Dati un click pe butonul marcat cu semnul “...” din partea dreaptă a evenimentului.

În acest fel veți deschide pagina de cod VBA la procedura și evenimentul dorit și tot ce vă mai rămâne de făcut este să modificați sau să introduceți codul VBA ca pe un simplu text. Access 97 oferă programatorilor pe măsura introducerii codului o informare rapidă asupra sintaxei comenzilor folosite, prin afișarea unei etichete sub comanda introdusă.

```
Private Sub Combo0_Click()
CInt|
 CInt(Expression) As Integer
End Sub
```

Eticheta afișată nu intervine în acțiunea programatorului și, în plus, dispare singură în momentul încheierii introducerii de la tastatură a comenzii – deci nu încercați să apăsați Esc pentru a o elimina.

## 4.4. Semnificația codului folosit în formularul creat

Pentru a face pe deplin funcțional formularul creat trebuie să creați în cod VBA instrucțiunile necesare. Dacă este pentru prima dată când concepeți un program, vă va fi aproape imposibil să finalizați aplicația. De aceea vă oferim codul VBA, așa cum ar trebui să apară în pagina de cod atașată formularului dumneavoastră, pe care va trebui să-l introduceți de la tastatură ca pe un simplu text.

Convenții:

Textele scrise cu caractere înclinate și precedate de semnul ‘ sunt comentarii la instrucțiunile care urmează.

Instrucțiunile VBA sunt scrise cu caractere îngroşate doar în paginile explicative care urmează, în practică ele vor apare în pagina de cod cu litere normale.

Liniile orizontale au fost folosite pentru marcarea terminării unei proceduri sau declarații.

Numerele afişate în fața liniilor de cod au fost folosite doar pentru marcarea începuturilor liniilor de rând – deoarece în anumite cazuri nu s-a reuşit crearea unor linii de cod de lungime egală cu a paginii noastre.

1. **Option Compare Database** *'comenzi obligatorii pentru începutul unei*
2. **Option Explicit** *'pagini de cod*
3. **Public** TipControl **As** Object *'Va fi folosită pentru determinarea tipului de control selectat de utilizator*
4. **Public** NumeControl **As** String *'Pentru determinarea controlului selectat de utilizator*
5. **Public** LocalitateNoua **As** String *'Pentru stocarea temporară a numelui unei localități noi*
6. **Public** Comuna **As** String *'Pentru stocarea temporară a numelui unei comune*
7. **Public** CodLocalitate **As** String *'Pentru stocarea temporară a codului localității*
8. **Public** NumeJudetNou **As** String *'Pentru stocarea temporară a numelui unui județ nou creat*
9. **Public** PrescJudetNou **As** String *'Pentru stocarea temporară a prescurtării unui județ nou creat*
10. **Public** Formular **As** Form *'Pentru determinarea formularului curent.*
11. **Public** TipSortare **As** String *'Pentru determinarea tipului de sortare.*
12. **Public** x **As** Integer *Variabilă temporară vizibilă în toate procedurile.*
13. **Private Sub** BtnTipareste_Click() *'Declar procedura ale cărei linii de cod vor fi lansate în execuție la executarea unui click pe controlul cu numele BtnTipareste.*
14. **Dim** Raspuns, Text *'Declar două variabile valabile doar în procedura curentă.*

15. Text = "1. La imprimantă." & **Chr**(13) *'Atribui expresia dintre ghilimele variabilei Text și îi atașez un sfârșit de linie.*
16. Text = Text & "2. Transfer în Word." & **Chr**(13) *'Adaug o altă expresie variabilei Text.*
17. Text = Text & "3. Transfer în Excel." *'Dacă după această nouă atribuire se 'va încerca afișarea variabilei Text aceasta va arăta astfel:*

    *1. La imprimantă.*
    *2. Transfer în Word.*
    *3. Transfer în Excel.*
18. Raspuns = **InputBox**(Text, "Alegeți numărul opțiunii...", 1) *'Atribui valoarea 'care va fi introdusă de utilizator în câmpul de editare al casetei cu titlul Alegeți 'numărul opțiunii ... și mesajul explicativ dat de conținutul variabilei Text, variabilei 'Raspuns*
19. **Select Case** Raspuns *'Deschid un ciclu de selecție bazat pe valoarea 'variabilei Raspuns*
20. **Case** 1 *'Dacă var. Raspuns are valoare 1(s-a ales opțiunea La 'imprimantă) atunci:*
21. **DoCmd.PrintOut** acPrintAll, , , acDraft, 1 *'Va fi transferat la imprimantă tot 'conținutul fișierului sursă al formularului curent.*
22. **Case** 2 *'Dacă var. Raspuns are valoarea 2 atunci:*
23. **DoCmd**.TransferSpreadsheet acExport, acSpreadsheetTypeLotusWK3, "tblJudete", "C:\My Documents\Judete.wk3" *'Conținutul fișierului tblJudete va fi exportat în 'fișierul Judete.wk3 din folderul My Documents.*
24. **Case** 3 *Dacă var. Raspuns are valoarea 3 atunci:*
25. **DoCmd**.TransferSpreadsheet acExport, acSpreadsheetTypeExcel97, "tblLocalitati", "C:\My Documents\ Localitati.xls" *'Conținutul fișierului tblLocalitati va fi 'exportat într-un fișier Excel 97 cu numele Localitati.xls, în folderul My Documents*

26. **Case** Else *'Dacă var. Raspuns are o valoare diferită de 1 sau 2 sau 3 'atunci:*
27. **MsgBox** "Trebuie să alegeți o opțiune de transfer!" *'Va fi afişată o casetă de 'avertizare cu mesajul specificat.*
28. **End Select** *'Închei ciclul de selecție.*
29. **End Sub** *'Închei execuția procedurii.*
30. **Private Sub** CboTipsortare_Click() *'Declar o nouă procedură ale cărei 'instrucțiuni vor fi executate în momentul executării unui click pe controlul 'CboTipSortare.*
31. **Select Case** CboTipsortare *'Deschid un ciclu de selecție a valorii controlului 'CboTipSortare.*
32. **Case** CboTipsortare = "Ascendent" *'Dacă acesta are valoarea Ascendent atunci:*
33. TipSortare = "**ASC**" *'Variabila publică TipSortare primeşte valoarea ASC.*
34. **Case** CboTipsortare = "Descendent" *'Iar dacă are valoarea Descendent atunci:*
35. TipSortare = "**DESC**" *'Var. TipSortare primeşte valoarea DESC.*
36. **End Select** *'Închei ciclul de selecție.*
37. **End Sub** *'Închei execuția procedurii.*
38. **Private Sub** Form_Load() *'Procedură care va fi executată la deschiderea 'formularului.*
39. ListaJudete = 18 *'Controlul ListaJudete primeşte valoarea 18, care corespunde în 'baza noastră de date județului Dolj*
40. ListaJudete_Click *'Se forțează lansarea în execuție a procedurii cu numele 'ListaJudete_Click-*
41. **End Sub** *'Închei execuția procedurii.*

---

42. **Private Sub** ListaJudete_Click() *'La executarea unui click pe controlul 'ListaJudete.*

43. NumeControl = ListaJudete. Name *'variabila NumeControl preia numele 'controlului ListaJudete*

44. ListaJudete.StatusBarText = "Județul" & **Format**(ListaJudete.Column(2), ">") *'Pe bara de status a formularului se afişează expresia Judetul şi numele județului 'selectat scris cu majuscule. Controlul ListaJudete are 3 coloane, însă în Access 97 'numerotarea coloanelor începe de la 0, astfel că proprietatea Column(2) înseamnă de 'fapt coloana a 3-a a controlului.*

45. ListaLocalitati.RowSource = "**SELECT DISTINCTROW** tblLocalitati.IdLocal" _
& "itate, tblLocalitati.Localitate, tblLocalitati.Comuna, tblLocalitati.CodLoc" _
& "alitate, tblLocalitati.IdJudet **FROM** tblLocalitati **WHERE**" _
& " (((tblLocalitati.IdJudet)=[Forms]![Form1]![ListaJudete]));"
*'Definesc sursa înregistrărilor controlului ListaLocalitati şi pun condiția selectării doar a 'acelor localități din tabelul tblLocalitati al căror cod de identificare al județului are 'valoarea egală cu valoarea controlului ListaJudete din formularul Form1. După cum se 'observă linia 45 este compusă de fapt din patru linii de rând, care deşi au fost încheiate cu 'Enter sunt interpretate de Access 97 ca fiind una singură. Acest lucru este posibil în 'Access '97 datorită prezenței combinației " _ la sfârşitul rândului şi prezenței & " la 'începutul 'următorului rând. Metoda este deosebit de utilă atunci când se lucrează pe un 'monitor cu o rezoluție foarte mică..*

46. **End Sub** *'Închei execuția procedurii.*

47. **Private Sub** ListaLocalitati_Click() *'La executarea unui click pe controlul 'ListaLocalitati.*

48. NumeControl = ListaLocalitati.Name *'variabila NumeControl preia numele 'controlului ListaJudete*

49. **If** ListaLocalitati.Column(2) = "" **Then** *'Dacă numele localității (preluat din coloana 'a 3-a a controlului ListaLocalitati este nul – nu a fost selectat nimic – atunci:*

50. ListaLocalitati.StatusBarText = "Localitatea " _
& **Format**(ListaLocalitati.Column(1), ">") & " " & "Cod Poştal " _
& **Format**(ListaLocalitati.Column(3), ">") & " " & "Județul " _
& **Format**(ListaJudete.Column(2), ">")

*'Pe bara de status a formularului curent va fi afişată expresia Localitatea căreia îi vor 'fi ataşate: 'numele localității cu majuscule şi expresia Cod Poştal*
*'codul Poştal cu majuscule şi expresia Județul*
*'numele județului cu majuscule*

51. **Else** *'În caz contrar (a fost selectat un nume din listă)*

52. ListaLocalitati.StatusBarText = "Localitatea " _
& **Format**(ListaLocalitati.Column(1), ">") & " " _
& "Comuna " & **Format**(ListaLocalitati.Column(2), ">") & " " & _
&"Cod Poştal " & **Format**(ListaLocalitati.Column(3), ">") & " " & _
& "Județul " & **Format**(ListaJudete.Column(2), ">")

*'Pe bara de status a formularului curent va fi afişată expresia Localitatea căreia îi 'vor fi ataşate:*
*'numele localității cu majuscule*
*'expresia Comuna şi numele comunei cu majuscule*
*'expresia Cod Poştal şi codul poştal cu majuscule*
*'expresia Județul şi numele județului cu majuscule*

53. **End If** *'Închei ciclul de analiză*
54. **End Sub** *'Închei execuția procedurii.*
55. **Private Sub** ListaLocalitati_DblClick(Cancel As Integer) *'La execuția unui dublu 'click pe controlul ListaLocalitati:*
56. **If** ListaSelectii.ListCount = 0 **Then** *'Dacă în controlul ListaSelectii nu se află nici 'un element, atunci:*
57. ListaSelectii.RowSource = ListaLocalitati.Column(1) *'Sursa înregistrărilor 'controlului ListaSelectii este formată dintr-un singur element identic cu expresia 'valoarea de pe coloana a 2-a a controlului ListaLocalitati.*
58. **Else** *'În caz contrar(dacă în ListaSelectii există cel puțin un element) atunci:*
59. ListaSelectii.RowSource = ListaSelectii.RowSource & ";" & ListaLocalitati.Column(1)
60. **End If** *'Închei ciclul de analiză.*
61. **End Sub** *'Închei execuția procedurii.*
62. **Private Sub** ListaSelectii_Click() *'Dacă se execută un click pe controlul 'ListaSelectii*
63. NumeControl = ListaSelectii.Name *'Var. NumeControl preia numele controlului 'ListaSelectii*
64. x = ListaSelectii.ListCount *'Var. x devine egală cu numărul de elemente afișate în 'controlul ListaSelectii.*
65. **End Sub** *'Închei execuția procedurii.*
66. **Private Sub** BtnAdauga_Click() *'La executarea unui click pe butonul cu 'numele BtnAdauga.*
67. **Adaug** *'Forțez executarea procedurii Adaug*
68. **End Sub** *'Închei execuția procedurii.*
69. **Private Sub** BtnSterge_Click() *'La executarea unui click pe butonul cu 'numele BtnSterge .*
70. **Sterg** *'Forțez executarea procedurii Sterg.*
71. **End Sub** *'Închei execuția procedurii.*
72. **Private Sub** BtnSorteaza_Click() *'La executarea unui click pe butonul 'BtnSorteaza.*
73. **Dim** n, m **As** Integer *'Definesc variabilele n și m de tip Integer*

74. **Dim** Temp **As** String *'Definesc variabila Temp de tip şir.*

75. **ReDim** Var(x) *'Redefinesc şirul de lungime x*

76. **Select Case** NumeControl *'deschid un ciclu de selecţie bazat pe valoarea 'variabilei NumeControl*

77. **Case** "ListaJudete" *'Dacă var. NumeJudete = ListaJudete (expresie!) atunci:*

78. **If** CboTipsortare = "Ascendent" **Then** *'Dacă controlul CboTipSortare are valoarea 'Ascendent, atunci:*

79. ListaJudete.RowSource = "**SELECT DISTINCTROW** tblJudete.IdJudet, tblJudet" _
& "e.Prescurtare, tblJudete.NumeJudet **FROM** tblJudete **ORDER BY** tblJudet" _
& "e.NumeJudet **ASC**;"

*'Sursa înregistrărilor controlului ListaJudete va fi refăcută din înregistrările tabelului 'tblJudet ordonate alfabetic.*

80. **Else** *'Dacă însă NumeJudete are altă valoare de cât Ascendent atunci:*

81. ListaJudete.RowSource = "**SELECT DISTINCTROW** tblJudete.IdJudet, tblJudet" _
& "e.Prescurtare, tblJudete.NumeJudet **FROM** tblJudete **ORDER BY** tblJudet" _
& "e.NumeJudet **DESC**;"

*'Sursa înregistrărilor controlului ListaJudete va fi refăcută din înregistrările tabelului 'tblJudet ordonate invers alfabetic.*

82. **End If** *'Închei analiza controlului NumeJudete.*

83. **Case** "ListaLocalitati" *'Dacă NumeControl are valoarea ListaLocalitati:*

84. **If** CboTipsortare = "Ascendent" **Then** *'şi dacă tipul de sortare = Ascendent atunci:*

ListaLocalitati.RowSource = "**SELECT DISTINCTROW** tblLocalit" _
& "ati.IdLocalitate, tblLocalitati.Localitate, tblLocalitati.Comuna, tblLocal" _
& "itati.CodLocalitate, tblLocalitati.IdJudet **FROM** tblLocalitati " _
& "**WHERE** (((tblLocalitati.IdJudet)=" & ListaJudete & "))" _
& " **ORDER BY** tblLocalitati.Localitate **ASC**;"

*'Sursa înregistrărilor controlului ListaLocalitati va fi refăcută din înregistrările*

*'tabelului tblLocalitati ai căror identificatori de județ sunt egali cu valoarea implicită '(Coloana 0) a elementului selectat în controlul ListaJudete .ordonate crescător.*

85. **Else** *'În caz contrar:*

ListaLocalitati.RowSource = "**SELECT DISTINCTROW** tblLocalit" _
& "ati.IdLocalitate, tblLocalitati.Localitate, tblLocalitati.Comuna, tblLocal" _
& "itati.CodLocalitate, tblLocalitati.IdJudet **FROM** tblLocalitati " _
& "**WHERE** (((tblLocalitati.IdJudet)=" & ListaJudete & "))" _
& " **ORDER BY** tblLocalitati.Localitate **DESC**;"

*'Sursa înregistrărilor controlului ListaLocalitati va fi refăcută din înregistrările 'tabelului tblLocalitati ai căror identificatori de județ sunt egali cu valoarea implicită '(Coloana 0) a elementului selectat în controlul ListaJudete .ordonate descrescător.*

86. **End If**
87. **Case** "ListaSelectii" *'Dacă var. NumeControl are valoarea ListaSelectii atunci:*
88. **If** CboTipsortare = "Ascendent" **Then** *'Dacă tipul de sortare a fost stabilit la 'valoarea Ascendent atunci:*
89. **For** n = 0 **To** ListaSelectii.ListCount - 1
90. Var(n) = ListaSelectii.ItemData(n) *'Elementul n al șirului Var(n) preia valoarea 'elementului n din lista ListaSelectii.*
91. **Next** n
92. **For** n = 0 **To** ListaSelectii.ListCount - 1
93. **For** m = 1 + n **To** ListaSelectii.ListCount - 1
94. **If** Var(n) > Var(m) **Then** *'Dacă elementul n este mai mare decât elementul m*
95. Temp = Var(n) *'Var. Temp preia valoarea elem. n*
96. Var(n) = Var(m) *'Elementul n preia valoarea elementului m*
97. Var(m) = Temp *'Elementul m preia valoarea variabilei Temp.*
98. **End If**
99. **Next** m *'Se reia ciclul de transferuri și verificări pentru următoare valoare a lui 'm*

```
Next n 'Se reia ciclul de transferuri şi verificări pentru următoarea valoare a lui 'n.
ListaSelectii.RowSource = "" 'Anulez sursa înregistrărilor controlului ListaSelectii.
Temp = "" 'Anulez valoarea var. Temp.
For n = 0 To m - 1
Temp = Temp & Var(n) & ";" 'În variabil Temp se încarcă toate valorile elementelor 'şirului Var(n) separate prin semnul ;
Next n 'Încarc următoare valoare a lui n.
Temp = Left(Temp, Len(Temp) - 1) 'Extrag ultimul caracter (semnul ;) din Temp
ListaSelectii.RowSource = Temp 'Dat fiind că sursa înregistrărilor controlului 'ListaSelectii este compusă din valorile unor variabile transformate într-un şir de elemente separate prin semnul ; folosesc şirul Temp ca sursă a înregistrărilor.
Else 'Dacă s-a ales o sortare de tip descrescător atunci:
For n = 0 To ListaSelectii.ListCount - 1
Var(n) = ListaSelectii.ItemData(n) 'Preiau valorile din controlul ListaSelectii în 'şirul Var(n).
Next n
For n = 0 To ListaSelectii.ListCount - 1
For m = 1 + n To ListaSelectii.ListCount - 1
If Var(n) < Var(m) Then
Temp = Var(n) 'Sortez descrescător valorile şirului.
Var(n) = Var(m) '...
Var(m) = Temp '...
End If
Next m
Next n
ListaSelectii.RowSource = "" 'Anulez sursa înregistrărilor controlului ListaSelectii.
Temp = "" 'Anulez valoarea var. Temp.
```

123. **For** n = 0 **To** m - 1

124. Temp = Temp **&** Var(n) **&** ";" *'Construiesc şirul Temp din elementele şirului Var(n).*

125. **Next** n

126. Temp = **Left**(Temp, **Len**(Temp) - 1) *'Extrag ultimul caracter (semnul ;)*

127. ListaSelectii.RowSource = Temp

128. **End If**

129. **End Select** *'Închei ciclul de analiză*

130. **End Sub** *'Închei execuția procedurii.*

---

131. **Function** Adaug() *'Definesc funcția Adaug*

132. **Dim** Selectie **As** String *'Definesc variabila Selectie*

133. **If** NumeControl = "ListaJudete" **Then** *'Dacă var. NumeControl = ListaJudete:*

134. NumeJudetNou = **InputBox**("Introduceți numele noului județ", "Județ nou!", "NUMELE JUDETULUI")

*'Var. NumeJudetNou preia valoarea introdusă de la tastatură de utilizator.*

135. PrescJudetNou = **InputBox**("Introduceți prescurtarea noului județ", "Județ nou!", "PRESCURTAREA JUDETULUI") *'Var. PrescJudetNou preia valoarea introdusă de la tastatură de utilizator.*

136. PrescJudetNou = **Format**(PrescJudetNou, ">") *'convertesc expresia din var. 'PrescJudetNou în majuscule*

137. CapitalizareText (NumeJudetNou) *'Inițializez funcția CapitalizareText cu 'argumentul NumeJudetNou.*

138. **DoCmd.RunSQL** "**INSERT INTO** tblJudete (Prescurtare, NumeJudet) **SELECT** " _

**&** PrescJudetNou **&** " **AS** Expr1, " **&** Text **&** " **AS** Expr2;"

*'Lansez o interogare de adăugare în tabelul tblJudete a datelor preluate de la 'tastatură. Astfel valoarea var. PrescJudetNou va fi*

*adăugată în câmpul Prescurtare, 'iar valoarea var. Text (valoarea var. NumeJudetNou convertită în majuscule) o adaug 'în câmpul NumeJudet.*

139. **ElseIf** NumeControl = "ListaSelectii" **Then** *'Dacă NumeControl = ListaSelectii*

140. **If** ListaLocalitati.Value = "" **Then** *'şi dacă nu este selectată nici o localitate*

141. **MsgBox** "Selectaţi o localitate!", vbExclamation, "Vă rugăm ..." *'Afişez o casetă de avertizare cu titlul Vă rugăm ... şi mesajul Selectaţi o localitate ...*

141. **End** *'Întrerup forţat execuţia procedurii*

142. **Else** *'Dacă însă a fost selectată o localitate în ListaLocalitati:*

143. **If** ListaSelectii.ListCount = 0 **Then** *'Dacă ListaSelectii nu conţine nici un element*

144. Selectie = **InputBox**("Introduceţi numele localităţii!", "Vă rugăm ... ", , , , , ListaLocalitati.Column(1)) *'Var. Selectie va prelua valoarea introdusă de utilizator de la tastatură.*

145. **If IsNull**(Selectie) **Then** *'Dacă s-a preluat o valoare nulă (s-a apăsat pe Cancel sau nu s-a introdus nimic şi s-a apăsat pe OK)*

146. **End** *'Forţez întreruperea execuţiei procedurii.*

147. **Else** *'Dacă s-a preluat o valoare însă*

148. ListaSelectii.RowSource = Selectie *'Sursa înregistrărilor controlului ListaSelectii se constituie din valoarea var. Selectie.*

149. **End If**

150. **Else** *'Dacă ListaSelectii conţine cel puţin un element atunci:*

151. Selectie = **InputBox**("Introduceţi numele localităţii!", "Vă rugăm ... ", ListaLocalitati.Column(1))

152. **If IsNull**(Selectie) **Then**

153. **End**

154. **Else**
155. ListaSelectii.RowSource = ListaSelectii.RowSource **&** ";" & Selectie *'Sursa 'înregistrărilor controlului ListaSelectii se constituie din valoarea vechilor înregistrări 'la care se adaugă valoarea var. Selectie separată de cele vechi cu ajutorul semnului ;*
156. **End If**
157. **End If**
158. **End If**
159. **ElseIf** NumeControl = "ListaLocalitati" **Then**
160. **End If**
161. LocalitateNoua = InputBox("Introduceți numele noii localități!", "Localitate Nouă!", "NUME LOCALITATE")
162. CapitalizareText (LocalitateNoua)
163. Comuna = **InputBox**("Introduceți numele comunei de care aparține localitatea " _
164. **&** Text & "1", "Comuna ...", "COMUNA")
165. CodLocalitate = **InputBox**("Introduceți codul poştal al localității " **&** Text & "!", "Cod Poştal...", "COD POŞTAL")
166. **If** LocalitateNoua = "" **Then**
167. **MsgBox** "Nu ați specificat numele localității! Reluați operațiunea!", vbCritical, "Eroare!"
168. **End If**
169. **If** Comuna = "" Then
170. **MsgBox** "Nu ați specificat numele comunei! Reluați operațiunea!", vbInformation, "Eroare!"
171. **End If**
172. **If** codpostal = "" **Then**
173. **MsgBox** "Nu ați specificat codul poştal al localității! Atribuim valoarea 1.", vbQuestion, "Eroare!"
174. **End If**
175. **'INSERT INTO** tblJudete (Prescurtare, NumeJudet) **SELECT** [nnn] **AS** Expr1, [mmm] AS Expr2 **FROM** tblJudete;
176. **End Function** *'Închei execuția funcției curente.*

```
Function Sterg()
Dim Selectie As String, SursaNoua, Item
If NumeControl = "ListaJudete" Then
MsgBox "Nu este permisă ştergerea numelui judeţului!", vbCritical,
                                    "Operaţie interzisă!"
End
ElseIf NumeControl = "ListaSelectii" Then
Selectie = ListaSelectii.ListIndex 'Var. Selectie devine egală cu numărul
                                    elementului selectat în controlul ListaSelectii.
For Item = 0 To ListaSelectii.ListCount   'De la 0 la numărul de elemente
                                    al controlului 'ListaSelectii.
If ListaSelectii.ItemData(Item) <> ListaSelectii.Value Then        'Dacă
                                    valoarea elementului n din ListaSelectii este
                                    diferită de valoarea elementului selectat din
                                    ListaSelectii atunci:
SursaNoua = SursaNoua & ListaSelectii.ItemData(Item) & ";"       'La
                                    valoarea var. 'SursaNoua adaug elementul n
                                    al controlului ListaSelectii şi semnul ;
End If
Next Item
ListaSelectii.RowSource = ""
ListaSelectii.RowSource = SursaNoua
ElseIf NumeControl = "ListaLocalitati" Then
DoCmd.RunSQL "DELETE tblLocalitati.*, tblLocalitati.IdLocalitate
                                    FROM " _
& " tblLocalitati WHERE (([tblLocalitati]![IdLocalitate]=" _
& ListaLocalitati.Column(0) & "));"
                                    'Lansez o interogare de ştergere a câmpurilor
                                    din tabelul tblLocalitati al căror 'identificator
                                    este egal cu valoarea de pe prima coloană a
                                    elementului selectat în 'controlul
                                    ListaLocalitati.
ListaLocalitati.Requery
End If
End Function
```

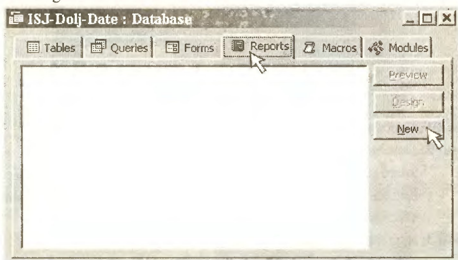
```
Function CapitalizareText(Text As String)
Dim n As Integer
For n = 1 To Len(NumeJudetNou)
If n = 1 Then
Text = Format(Left(NumeJudetNou, 1), ">")
Else
Text = Text & Format(Mid(NumeJudetNou, n, 1), "<")
End If
Next n
End Function
```

# RAPOARTE

Prezentarea şi prelucrarea datelor .cu ajutorul formularelor este extrem de utilă în activitatea practică, însă pentru crearea unei aplicaţii complexe, care să satisfacă pe deplin cerinţele utilizatorilor, este important să permiteţi şi tipărirea datelor într-o formă elegantă. Pentru aceasta este necesară înţelegerea modului în care funcţionează rapoartele Access 97.

## 5.1. Crearea rapoartelor

Cea mai simplă cale petru crearea unui raport este folosirea unuia din asistenţii Wizard (Report Wizard) pentru crearea rapoartelor. Dacă fereastra Database este vizibilă în fereastra aplicaţiei Access 97 executaţi un click pe eticheta Reports astfel încât această să apară în prim plan aşa cum se observă în figura următoare:

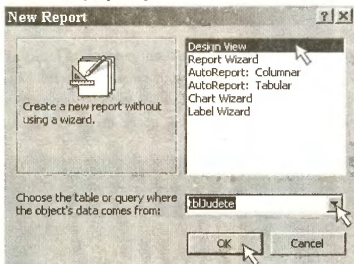


Dacă în baza dumneavoastră de date nu ați creat anterior nici un raport, butoanele Preview și Design vor fi dezactivate. Apăsați pe butonul New pentru a lansa în execuție asistentul Wizard pentru crearea rapoartelor.

### 5.1.1. Modul Design View

Este modul în care utilizatorului îi revine sarcina realizării integrale a unui raport. Chiar dacă acest modalitate de creare nu implică acțiunea asistenților Wizard, aceasta este modalitatea care oferă cele mai multe posibilități programatorului – deși solicită un volum mare de cunoștințe din partea acestuia.

Dacă în figura anterioară ați apăsat pe butonul New, în ecranul următor va trebui să precizați modalitatea de crearea a noului raport, sursa înregistrărilor acestuia și tipul raportului.

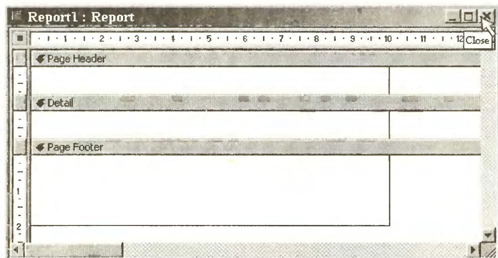



**Pasul 1:** Executați un click pe opțiunea Design View.

**Pasul 2:** Deschideți lista combinată care conține lista tabelelor și interogărilor bazei de date și alegeți tabelul sau interogarea care va furniza datele necesare raportului dumneavoastră. Alegeți ca sursă a înregistrărilor tabelul tblJudete.

**Pasul 3:** Executați un click pe butonul OK pentru a putea trece la pasul următor.

Acest mod de creare a unui raport, la fel ca şi modul Report Wizard, nu impune obligativitatea declarării unei surse a înregistrărilor, deoarece veţi avea ulterior posibilitatea să declaraţi sau să creaţi o nouă sursă de înregistrări pentru raportul dumneavoastră.




Odată cu apariţia imaginii de mai sus pe ecranul calculatorului dumneavoastră, acţiunea asistentului Wizard pentru crearea rapoartelor ia sfârşit, programatorul trebuind să decidă singur asupra datelor şi formei datelor care vor apare în raport. Întrucât completarea datelor acestui raport este o operaţiune complexă ea va fi dezvoltată în capitolul 5.2.5. Închideţi raportul executând un click pe butonul Close – aşa cum se observă în figură.

Acceptaţi salvarea raportului creat, chiar dacă deocamdată, acesta nu conţine nimic şi apăsaţi pe butonul Yes. Denumiţi acest raport RpJudeteDV şi apăsaţi pe butonul OK pentru finalizarea operaţiunii de creare a raportului.



### 5.1.2. Modul Report Wizard

Reprezintă modalitatea cea mai spectaculoasă de creare a unui raport cu ajutorul unui asistent Wizard. Dacă ați realizat pașii impuși în fig. 1 din acest capitol , executați un click pe opțiunea Report Wizard, așa cum se obesrvă în fig. 5-1-2-1. La fel ca și în cazul modului Design View, nu este obligatorie, în această fază precizarea tabelului care constituie sursa înregistrărilor raportului dumneavoastră, deoarece o veți putea stabili în ecranul următor după apăsarea pe butonul OK.

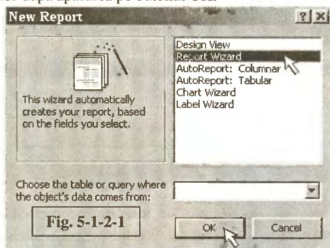


Fig. 5-1-2-1

În fig. 5-1-2-2 va trebui să precizați:

1. În caseta Table/Queries: numele tabelului care conține înregistrările necesare raportului dumneavoastră. (Alegeți tabelul tblLocalitati)

2. În lista Selected Fields numele câmpurilor din tabelul sursă care vor apare în raportul dumneavoastră. Pentru introducerea unui câmp în lista Selected Fields, selectați-l în lista Available Fields, și executați un dublu click pe acesta sau un click pe butonul marcat cu semnul >.

Dacă raportul dumneavoastră trebuie să conțină date și din alte tabele sau interogări ale bazei de date reluați pașii 1 și 2, pornind de la un alt tabel sau interogare sursă (În exemplul nostru au mai fost preluate câmpurile Prescurtare și NumeJudet din tabelul tblJudet). Atenție: în cazul folosirii mai multor tabele sau interogări sursă pentru crearea unui raport sau

formular, este obligatoriu ca între tabelele sau interogările folosite să fie definită o relație.

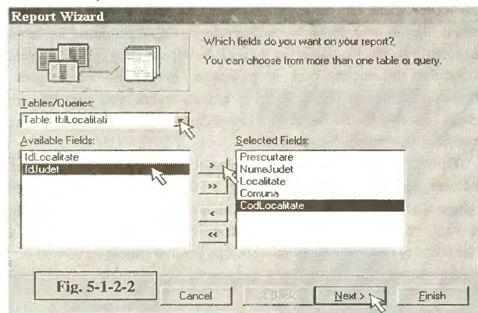



Fig. 5-1-2-2

Report Wizard

How do you want to view your data?

by tblJudete
by tblLocalitati

Show me more information

Prescurtare, NumeJudet

Localitate, Comuna, CodLocalitate

Cancel | < Back | Next > | Finish

Fig. 5-1-2-3

În fig. 5-1-2-3 va trebui să stabiliți modul în care va fi realizată sortarea datelor din tabelele sursă în cadrul raportului dumneavoastră. Astfel dacă veți executa un click pe opțiunea "by tblLocalitati" raportul va prezenta datele în ordinea în care acestea au fost introduse în tabelul tblLocalitati, iar dacă veți executa un click pe opțiunea "by tblJudete" datele din tabelul tblLocalitati vor fi prezentate în ordinea cronologică a introducerii, sortate pe județe (la fel în ordinea cronologică a introducerii). Sortarea datelor în cadrul raportului este redată destul de sugestiv în caseta de previzualizare din partea dreaptă.

Gruparea datelor în cadrul raportului va fi realizată în fereastra următoare, așa cum este prezentată în fig. 5-1-2-4. Pentru stabilirea nivelelor de grupare, selectați din lista cu numele câmpurilor disponibile câmpul după care va fi realizată gruparea și transferați-l în partea dreaptă (caseta de previzualizare) prin executarea unui click pe butonul marcat cu semnul >.

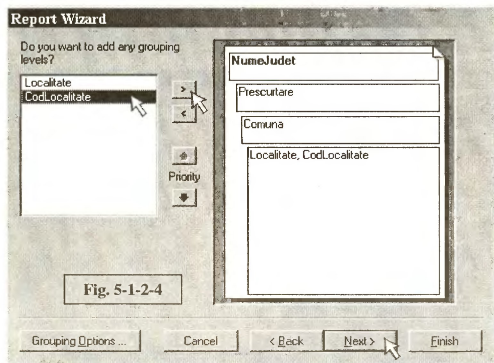



Fig. 5-1-2-4

În caseta de previzualizare nivelele de grupare vor fi marcate cu culoarea albastră iar informaţiile grupate în acestea cu culoarea neagră. În exemplul nostru localităţile înregistrate şi codul poştal al acestora vor fi grupate după comunele de care aparţin, iar acestea la rândul lor sunt grupate după judeţele din care fac parte.

Gruparea datelor poate fi realizată după iniţialele acestora (1 până la 5 litere) sau după numele întreg (varianta "Normal"). Stabilirea unei anumite variante de grupare (alta decât cea obişnuită –"normal") poate fi realizată prin apăsarea pe butonul Grouping Options.

Sortarea datelor din secţiunea "Detaliu" – datele din ultimul nivel de grupare stabilit – trebuie precizată, deoarece implicit asistentul Wizard consideră că datele raportului trebuiesc prezentate în ordinea cronologică a introducerii acestora, însă nu poate fi realizată decât pentru cel mult 4 câmpuri, aşa cum se observă din fig. 5-1-2-5. Alegerea unui câmp oarecare din lista câmpurilor disponibile presupune implicit sortarea ascendentă a datelor. Pentru o sortare descendentă, după alegerea câmpului în casetele de sortare, executaţi un click pe butonul de sortare din partea dreaptă a casetei

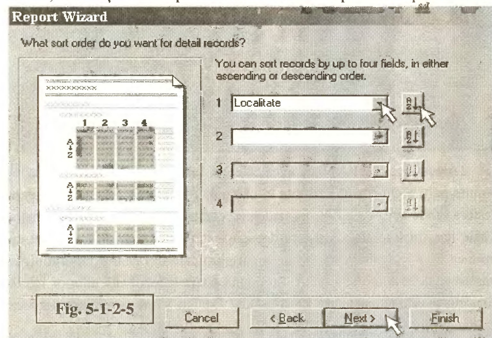



Fig. 5-1-2-5

În fig.5-1-2-6 trebuiesc precizate:

1. Tipul dispunerii datelor în cadrul raportului ce va fi tipărit – caseta Layout.

2. Modalitatea transpunerii raportului în pagină – caseta Orientation (Portrait – scriere normală, Landscape – pe lățime).

3. Dacă se doreşte trunchierea datelor astfel încât toate câmpurile selectate să fie prezente în cadrul raportului – caseta de validare "Adjust ".

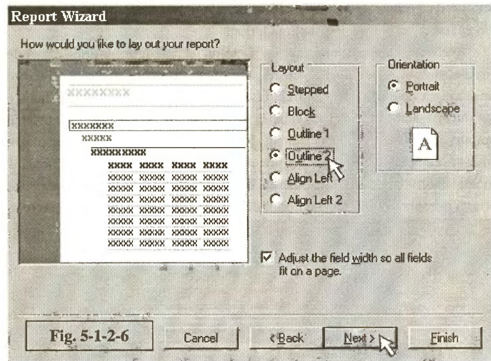


Fig. 5-1-2-6

Trunchierea datelor poate avea ca efect "ascunderea" la tipărire a unei anumite părți a datelor înregistrate în câmpurile selectate, iar ne-trunchierea acestora poate avea ca efect "ascunderea" completă a câmpurilor în situația în care lungimea totală a acestora depăşeşte lungimea paginii. Indiferent de opțiunea dumneavoastră datele nu vor fi eliminate din cadrul raportului, ci doar ascunse în anumite situații. Problema poate fi rezolvată ulterior foarte uşor prin deschiderea raportului în modul Design View şi redimensionarea

manuală a mărimii câmpurilor afectate de una din situațiile prezentate mai sus.

Stabilirea mărimii şi tipului caracterelor utilizate poate fi rezolvată într-o manieră destul de elegantă în ecranul prezentat în fig.5-1-2-7. Access 97 vă pune la dispoziție câteva scheme preconfigurate pentru realizarea unui aspect grafic elegant al rapoartelor întocmite, pe care le veți putea modifica ulterior în modul Design View, în cazul în care nu îndeplinesc cerințele dumneavoastră.

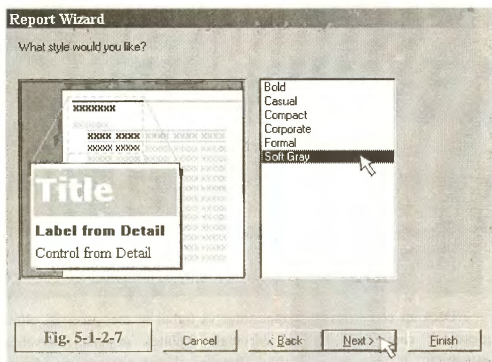


Fig. 5-1-2-7

Executarea unui click pe numele uneia din schemele preconfigurate, prezentate în lista din partea dreaptă a ecranului, va avea ca efect modificarea aspectului casetei de previzualizare din partea stângă şi prezentarea caracteristicilor de formatare pentru principalele elemente ale unui formular: Titlu, etichete explicative, elemente de detaliu.

Trecerea la ecranul următor (fig. 5-1-2-8) marchează sfârşitul intervenției asistentului Wizard pentru crearea rapoartelor.

Titlul raportului, aşa cum va apare la tipărire, pe care trebuie să-l introduceţi în câmpul “What title… “ va fi preluat şi folosit în acelaşi timp şi ca nume pentru raportul dumneavoastră, ceea ce poate constitui o problemă a asistentului Wizard, dar care poate fi corectată utlerior prin schimbarea denumirii raportului.

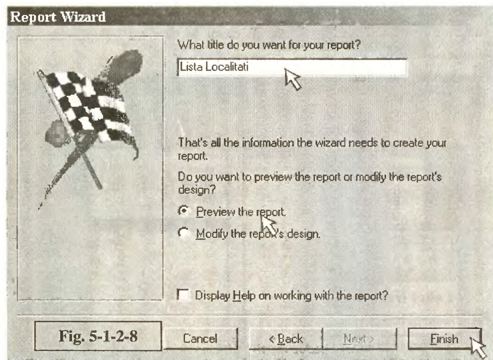


Fig. 5-1-2-8

Dacă aţi completat titlul raportului va trebui să mai precizaţi ultima operaţiune care doriţi să fie executată înaintea stopării intervenţiei asistentului: vizualizarea raportului (Preview the report) sau modificarea formei finale a raportului (Modify the report’s design).

Optaţi, pentru continuarea exemplului nostru, pentru vizualizarea raportului şi apăsaţi pe butonul Finish.

Forma finală a raportului, aşa cum va apare de acum înainte este prezentată în fig.5-1-2-9.

În situaţia în care această formă de prezentare a raportului nu îndeplineşte cerinţele dumneavoastră, o puteţi modifica apăsând pe butonul View de pe bara de unelte Print Preview.

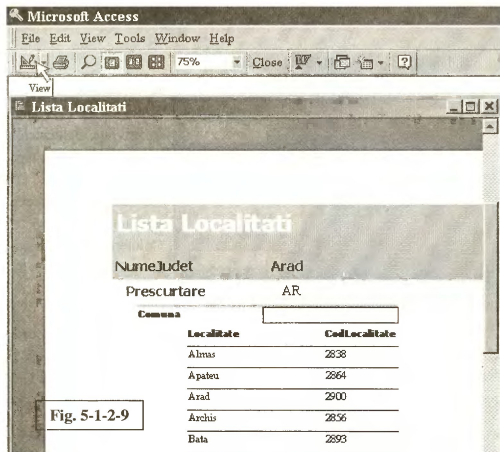


Fig. 5-1-2-9

Observând cu atenţie raportul creat veţi sesiza faptul că în câmpul afectat numelui comunei (sub prescurtarea judeţului) nu apare scris nimic. Explicaţia rezultă din faptul că:

➢ în tabelul tblLocalitati pe coloana Comuna în dreptul localităţilor respective nu a fost înregistrată nici o informaţie.

➢ sortarea datelor a fost realizată în ordine alfabetică atât la nivelul localităţilor cât şi al comunelor iar în logica Access 97 lipsa unei valori

oarecare dintr-un câmp presupune totuşi existenţa unei valori în acel câmp – şi anume valoarea nulă (Null).

➢ valoarea Null ocupă primul loc în cadrul unei liste de valori sortate ascendent.

### 5.1.3. Modul AutoReport: Columnar

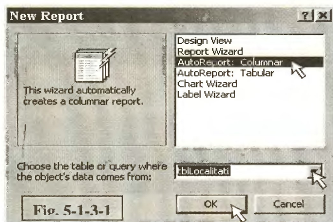



Fig. 5-1-3-1

Realizarea unui raport fără prea mari pretenţii în privinţa elementelor grafice şi prelucrării datelor poate fi realizată prin selectarea opţiunii AutoReport: Columnar din fereastra New Report.

Spre deosebire de variantele prezentate anterior, de această dată este obligatorie precizarea tabelului sursă. Dacă veţi alege ca în fig. 5-1-3-1 tabelul tblLocalitati ca sursă a înregistrărilor, după apăsarea pe butonul OK acesta va arăta astfel:

tblLocalitati

| | | | |
|---|---|---|---|
| 210 | Hateg. | 2650 | 22 |
| 294 | Nehoiu | 5194 | 11 |
| 195 | Golesti | 5316 | 41 |
| 196 | Grămesti | 5858 | 36 |
| 197 | Grânari | | 10 |
| 198 | Grădicesti | 5865 | 36 |
| 199 | Grebănu | 5267 | 11 |
| 200 | Gugesti | 5315 | 41 |
| 201 | Gura Humorului | 5900 | 36 |
| 202 | Gura Ocnitei | 217 | 17 |
| 204 | Gura Teghii | 5196 | 11 |
| 205 | Hălchiu | | 10 |
| 207 | Harghita Băi | 4122 | 23 |
| 190 | Geoagiu | 2616 | 22 |
| 209 | Hârtop | 5765 | 36 |

Fig. 5-1-3-2

### 5.1.4. Modul AutoReport: Tabular

Realizarea unui raport în care datele să fie dispuse sub forma unui tabel (nu sub forma unei coloane, ca în exemplul anterior) o puteți realiza prin selectarea opțiunii AutoReport: Tabular.

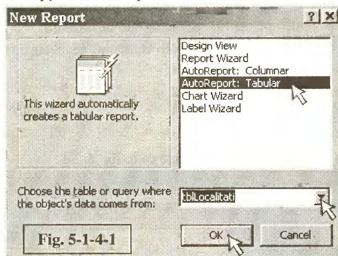


Fig. 5-1-4-1

Pentru a putea face comparația între cele două tipuri de rapoarte alegeți din nou tabelul tblLocalitati ca sursă a înregistrărilor și apăsați pe butonul OK. În fig.5-1-4-2 este prezentată forma raportului creat.



Fig. 5-1-4-2

## 5.2. Modificarea rapoartelor

Realizarea rapoartelor cu ajutorul asistenților Wizard este, în general destul de comodă, însă nu rezolvă toate aspectele necesare utilizatorilor. De aceea se impune, ca după realizarea unui astfel de raport, să-l modificați astfel încât să răspundă pe deplin cerințelor dumneavoastră.

Dacă în capitolele anterioare ați salvat rapoartele create cu numele:

| | |
|---|---|
| RpJudeteDV | cap. 5.1.1 |
| ListaLocalitati | cap. 5.1.2 |
| RpLocalitatiColumnar | cap. 5.1.3 |
| RpLocalitatiTabular | cap. 5.1.4 |

fereastra Database ar trebui să arate ca în fig. 5-2-1.

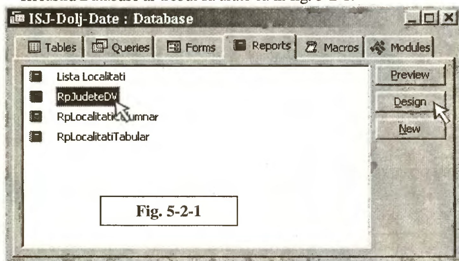



Fig. 5-2-1

Pentru modificarea raportului cu numele RpJudeteDV va trebui să:

**Pasul 1:** Efectuați un click pe numele său.

**Pasul 2:** Apăsați pe butonul Design din fereastra Database.

Odată intrați în interiorul ferestrei de proiectare va trebui să nu scăpați din vedere următoarele reguli care rezultă din schimbarea formei cursorului mouse-ului, în funcție de evenimentele provocate de dumneavoastră:

1. Trasarea unui dreptunghi, prin “agăţare şi tragere”, pe suprafaţa de lucru a raportului (sau formularului) permite selectarea tuturor obiectelor incluse în interiorul dreptunghiului şi a obiectelor pe care marginea acestuia le intersectează.

2. Permite redimensionarea orizontală (pe direcţia E-W!) a obiectului selectat.

3. Este permisă redimensionarea verticală (pe direcţia N-S!) a obiectului sau obiectelor selectate.

4. Este permisă redimensionarea pe diagonală (direcţia SW-NE!) a obiectului selectat.

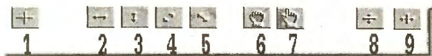
5. Este permisă redimensionarea pe diagonală (direcţia SE-NW!) a obiectului selectat.

6. Este permisă deplasarea unui obiect sau unui grup de obiecte selectate.

7. Este permisă deplasarea unui singur obiect din cadrul unui grup de obiecte legate.

8. Este posibilă redimensionarea verticală a secţiunii unui raport (sau formular).

9. Permite redimensionarea orizontală a secţiunii unui raport (sau formular).




Redimensionările care sunt posibile astfel cu ajutorul mouse-ului pot fi realizate şi prin modificările valorilor proprietăţilor Width şi Height ale obiectelor selectate, iar deplasările obiectelor în alte poziţii în cadrul unei secţiuni a formularului pot fi realizate prin modificarea valorilor câmpurilor de proprietăţi Left (distanţa faţă de marginea din stânga a secţiunii curente) şi Top (distanţa faţă de marginea superioară a secţiunii curente). Modificarea valorilor acestor proprietăţi deşi foarte exactă solicită însă un efort suplimentar de concentrare din partea programatorului, de aceea

recomandăm memorarea explicațiilor formelor cursorului mouse-ului și lucrul cu acesta pentru redimensionarea și mutarea obiectelor dintr-un formular sau raport.

### 5.2.1. Secțiunile unui raport

Raportul RpJudeteDV deschis în modul de lucru Design (proiectare) va arăta ca în figura 5-2-1-1. Analizând elementele acestuia veți sesiza ușor următoarele secțiuni:

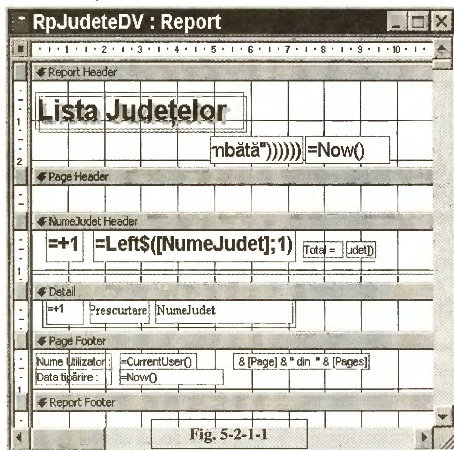



Fig. 5-2-1-1

1. Secțiunea Page Header (antet de pagină) – cuprinde elementele (de regulă elemente grafice) care vor apare în antetul fiecărei pagini a raportului dumneavoastră.

2. Secțiunea Detail (detalii) – cuprinde informațiile ce vor fi afişate în cadrul raportului.

3. Secțiunea Page Footer (subsol de pagină) – cuprinde informațiile ce vor fi afişate la sfârşitul fiecărei pagini a raportului (numărul paginii, data tipăririi, numele utilizatorului, etc.)

Afişarea unui alt grup de secțiuni posibile în cadrul unui raport, antetul şi subsolul raportului (Report Header, Report Footer) este posibilă prin selectarea opțiunii corespunzătoare din meniul View al aplicației Access 97. Informațiile conținute în aceste ultime secțiuni vor fi afişate o singură dată în cadrul formularului (la începutul – Report Header, sau la sfârşitul acestuia – Report Footer).

### 5.2.2. Adăugarea controalelor

De regulă asistenții Wizard ai aplicației Access 97 nu pot rezolva toate problemele solicitate de utilizator (afişarea tuturor câmpurilor necesare, poziționarea acestora în diferitele secțiuni ale raportului, dimensionarea sau formatarea câmpurilor, etc).

Adăugarea diferitelor controale necesare unui utilizator (câmpuri suplimentare, etichete explicative, imagini, casete sau liste combinate, etc.) va fi posibilă, în condițiile prezenței pe ecran a barei de unelte Toolbox, prin executarea unui click pe tipul controlului solicitat de pe bara de unelte menționată, urmat de trasarea cu ajutorul mouse-ului a unui dreptunghi, într-una din secțiunile raportului, care va marca suprafața în care va fi încadrat acel control în raportul dumneavoastră.

Dacă cele două operațiuni menționate în paragraful anterior vor fi efectuate atunci când butonul "Control Wizard" de pe bara de unelte Toolbox este "apăsat", veți beneficia de intervenția automată a altor asistenți Wizard specializați în crearea controalelor.

Adăugarea unui alt tabel care să constituie sursa înregistrărilor raportului (sau modificarea sursei înregistrărilor) poate fi realizată prin

executarea unui click (sau mai sigur a unui dublu click) în afara zonei ocupate de secțiunile raportului urmat de alegerea unui tabel (sau interogări) din caseta combinată "Record Source" a tabelului de proprietăți.

### 5.2.3. Gruparea înregistrărilor

Pentru a realiza gruparea datelor, în cadrul raportului, după diferite criterii definite de utilizator, deschideți meniul View al aplicației Access 97 executați un click pe opțiunea "Sorting and Grouping" (dacă iconița din partea stângă a acesteia nu este deja "apăsată") iar apoi în fereastra Sorting and Grouping alegeți numele câmpului după care va realizată sortarea sau gruparea datelor şi tipul de sortare.

Rolul câmpurilor ferestrei Sorting and Grouping este următorul:

Group Header - determină gruparea datelor în cadrul formularului şi afişarea unei secțiuni de antet corespunzătoare în cadrul raportului.(valoarea Yes)

Group Footer - afişarea secțiunii de subsol a grupării.

Group On - stabileşte modul de grupare a datelor (Each Value – după fiecare valoare conținută de câmpul selectat, Prefix Character – după primele n caractere ale valorilor conținute în acel câmp).

Group Interval - intervalul sau numărul caracterelor după care se face gruparea.

Keep Togheter - valoarea Yes forțează păstrarea datelor care compun un grup de înregistrări pe aceeaşi pagină a raportului.

## 5.3. Temă

Modificaţi raportul RpJudeteDv creat anterior astfel încât la final să arate ca în figura următoare:

RpJudeteDV : Report

Lista Județelor

Miercuri 29.08.2001

1 A Total = 3

1 AG ARGEŞ

2 AR ARAD

3 AB ALBA

Nume Utilizator : Admin Pagina 1 din 3

Data tipăririi : 29.08.2001 19:01:21

Page: 1

Elemente în sprijinul realizării temei:

Titlul raportului a fost creat prin suprapunerea a două etichete cu acelaşi conţinut şi tip de formatare dar cu fonturi de culori diferite.

Câmpul situat sub titlul raportul în care este afişată ziua curentă a fost realizat prin introducerea următoarei formule în câmpul de proprietăți "Control Source":

=**IIf(Weekday(Now()**=1;"Duminică";IIf(**Weekday(Now()**)=2;"Luni" ;**IIf(Weekday(Now()**)=3;"Marți";**IIf(Weekday(Now()**)=4;"Miercuri";**IIf( Weekday(Now()**)=5;"Joi";IIf(**Weekday(Now()**)=6;"Vineri";"Sâmbătă")))))

Numărarea automată a înregistrărilor a fost realizată prin introducerea expresiei "=+1" în câmpul de proprietăți Control Source şi stabilirea valorii "Over Group" pentru câmpul de proprietăți Running sum.

Numele utilizatorului care a facilitat tipărirea raportului a fost realizat prin introducerea formulei "=CurrentUser()" în câmpul de proprietăți Control Source.

Numărul paginii curente şi numărul total de pagini ale raportului a fost realizat prin introducerea formulei: ="Pagina " & **[Page]** & " din " & **[Pages]** în câmpul de proprietăți Control Source.

# MACROCOMENZI

Macrocomenzile reprezintă denumiri ale unei serii de acţiuni executate succesiv. Macrocomenzile pot funcţiona independent în cadrul bazei de date create sau pot fi ataşate unui control din cadrul formularului sau raportului creat, pentru a fi lansate la îndeplinirea unui anumit eveniment. În esenţă macrocomenzile sunt de fapt nişte proceduri publice predefinite.

## 6.1. Crearea unei macrocomenzi.

Pornind de la fereastra Database creată anterior, executaţi un click pe eticheta "Macros" urmat de un click pe butonul "New".

Dacă imaginea care va apare, în urma acestei acţiuni, nu va conţine coloanele "Macro Name" şi "Condition", executaţi câte un click pe butoanele "Macro Names" şi "Conditions" de pe bara de unelte, aşa cum se observă în figura anterioară.

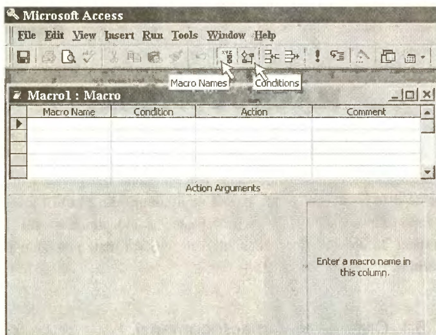
Rolul coloanelor:

**Macro Name** – conţine numele comenzii definite pe această cale de utilizator.

**Condition** – conţine elementele de condiţionare a executării comenzii definite.

**Action** – specifică acţiunea care va fi executată de comanda definită.

**Comment** – conţine explicaţii la rolul îndeplinit de comanda definită.



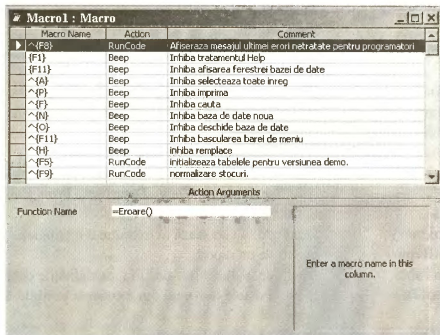

Completarea coloanelor macrocomenzii se face manual de către utilizator cu elementele necesare, cu excepția coloanei Action, a cărei completare presupune deschiderea casetei combinate și alegerea uneia din comenzile preconfigurate puse la dispoziție de aplicația Access 97.

Principalele acțiuni și rolul îndeplinit de acestea sunt redate în cele ce urmează:

| | |
|---|---|
| **AddMenu** | Adaugă un meniu în bara de meniu personalizată a unui formular sau raport. |
| **Beep** | Emite un sunet de avertizare. |
| **Cancel Event** | Întrerupe un eveniment specific Access 97. |
| **Close** | Închide o fereastră specificată. |
| **Copy Object** | Creează o copie a bazei de date. |
| **Delete Object** | Șterge un obiect specificat sau selectat. |
| **Echo** | Ascunde sau afișează rezultatul acțiunii unei macrocomenzi. |
| **FindRecord** | Caută prima sau următoare înregistrare care îndeplinește o anumită condiție. |

| | |
|---|---|
| **GoToRecord** | Efectuează un salt necondiţionat la o anumită înregistrare. |
| **Hourglass** | Schimbă forma implicită a cursorului într-una de forma unei clepsidre. |
| **Maximize** | Maximizează fereastra specificată sau implicită. |
| **Minimize** | Minimizează fereastra specificată sau implicită. |
| **MoveSize** | Mută o fereatră. |
| **MsgBox** | Determină apariţia unei casete de mesaj |
| **OpenForm** | Deschide un formular într-unul din modurile specifice: Design View, Datasheet, Print Preview, Form View. |
| **OpenQuery** | Deschide sau lansează în execuţie o interogare. |
| **OpenReport** | Deschide un raport. |
| **OpenTable** | Deschide un tabel într-una din modalităţile posibile. |
| **OutputTo** | Exportă datele într-unul din formatele posibile: RTF, txt, xls. |
| **PrintOut** | Tipăreşte obiectul specificat. |
| **Quit** | Închide aplicaţia Access 97. |
| **Rename** | Redenumeşte obiectul specificat. |
| **Requery** | Reactualizează o interogare sau un control definit de utilizator. |
| **RunApp** | Lansează în execuţia o aplicaţie externă. |
| **RunCode** | Lansează în execuţie o procedură VBA. |
| **RunCommand** | Execută o comandă a unui meniu. |
| **RunMacro** | Lansează în execuţie o macrocomandă. |
| **RunSQL** | Execută o interogare. |
| **Save** | Salvează obiectul definit sau selectat. |
| **SendKeys** | Simulează apăsarea anumitor taste în aplicaţia Access 97 sau în aplicaţia curentă. |
| **TransferText** | Realizează importul sau exportul unui fişier de tip text |

Încercaţi în exemplul următor să realizaţi o macrocomandă care îndeplineşte o funcţie specială:




Funcția Eroare care a fost introdusă în câmpul Function Name trebuie mai întâi definită ca fiind publică, într-un modul, pentru a putea fi folosită în cadrul acestei macrocomenzi.

Pentru ca acest macro să fie pe deplin funcțional redăm în tabelul de mai jos lista câmpurilor şi valorile atribuite acestora:

| Coloana Macro Name | Câmpul | Expresia introdusă |
|---|---|---|
| ^{F8} | Function Name | =Eroare() |
| ^{F5} | Function Name | =InitializareTabele() |
| ^{F5} | Function Name | =ReglareStocuri() |

Închideți şi salvați acest macro sub numele "Autoexec". Folosirea acestui nume vă asigură de faptul că la următoarea deschidere a bazei de date în care a fost introdusă această macrocomandă, vor fi lansate în execuție toate comenzile care o compun – cea ce în multe situații este extrem de necesar.

Din cuprins:

PREZENTAREA MENIURILOR
- Noțiuni generale
- Lansarea în execuție
- Fereastra Database
- Tabele (Table)

INTEROGĂRI (QUERY)
- Noțiuni generale
- Tipuri de interogări
- Crearea unei interogări
- Conversia interogărilor

FORMULARE (FORMS)
- Noțiuni generale
- Tipuri de controale
- Crearea unui formular

CODUL VBA
- Noțiuni generale
- Proceduri
- Instrucțiuni în cod VBA
- Adăugarea codului
- Semnificația codului folosit în formularul creat

RAPOARTE
- Crearea rapoartelor
- Modificarea rapoartelor

MACROCOMENZI
- Crearea unei macrocomenzi

COLECȚIA: INIȚIERE ÎN CALCULATOARE
NIVEL: ÎNCEPĂTORI, MEDIU

4

ISBN 973 - 85773 - 1 - 4

Preț: 60.000 lei